Survey of Afghan (Dari) Language NLP for Building Afghan NLIDB System

by

Sadullah Karimi
21166041

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
Masters of Science in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
September 2022

Declaration

It is hereby declared that

1. The thesis submitted is my original work while completing a degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. I have acknowledged all main sources of help.

Student's Full Name & Signature:

Sadullah Karimi
21166041

Approval

The thesis titled "Survey of Afghan (Dari) Language NLP for Building Afghan NLIDB System" submitted by Sadullah Karimi (21166041) of Fall 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Master's of Science in Computer Science and Engineering on (11/September/2022).

Examining Committee:
Supervisor (Member):

Annajiat Alim Rasel
Digitally signed by Annajiat Alim Rasel
DN: cn=Annajiat Alim Rasel, o=Brac University, ou=CSE Department, email=annajiat@bracu.ac.bd, c=BD
Date: 2022.09.18 08:05:23 +06'00'

Annajiat Alim Rasel, Senior Lecturer
Department of Computer Science and Engineering, Brac University

Faculty Member (Member):

Iqbal

Muhammad Iqbal Hossain, PhD, Associate Professor
Department of Computer Science and Engineering, Brac University

Faculty Member (Member):

Matin Saad Abdullah, PhD, Professor
Department of Computer Science and Engineering, Brac University

External Faculty Member (Member):

Mohammad Zahidur Rahman, PhD, Professor
Department of Computer Science and Engineering, Jahangirnagar Unversity

Program Coordinator (Member):

Amitabha Chakrabarty, PhD, Associate Professor
Department of Computer Science and Engineering, Brac University

Head of Department (Chair):

Sadia Hamid Kazi, PhD, Chairperson and Associate Professor
Department of Computer Science and Engineering, Brac University

Ethics Statement

Hereby, I Sadullah Karimi consciously assure that for the manuscript, Survey of Afghan (Dari) Language NLP for Building Afghan NLIDB System, the following is fulfilled:

1. This material is my original work, which has not been previously published elsewhere.
2. This research is currently not being considered for publication elsewhere.
3. This research reflects my research and analysis in a truthful and complete manner.
4. This research properly credits the meaningful contributions of the supervisor.
5. The results are placed in the context of prior and existing research.
6. The used sources have been cited or quoted.
7. The author has been personally and actively involved in substantial work leading to the paper and will take public responsibility for its content.

I agree with the all above statements and declare that this submission follows the policies and guidelines.

Dedication

I dedicate this research to Allah Jalla Jalāluhu who is my creator, my strong pillar, and my source of inspiration, wisdom, knowledge, and understanding. Throughout this program, Allah Jalla Jalāluhu has been the source of my strength, and with his support, I have soared. I also dedicate this work to my Mother, Eqlima Karimi, who has encouraged me all the way and whose encouragement has made sure that I give it all it takes to finish what I have started. To my respected supervisor, Mr. Annajiat Alim Rasel, who has really helped me to complete this research. Thank you, Dear Sir, God bless you.

Acknowledgement

I would like to take this opportunity to express my gratitude to those people who had given me the support and help throughout this research project and its implementation.

At first, I want to admit my gratefulness to Allah (ج) who enabled me to complete this work successfully. Then, I would like to give thanks to my thesis supervisor, Mr. Annajiat Alim Rasel, senior lecturer, Department of Computer Science and Engineering, Brac University for his continuous guidance, spontaneous support, relentless encouragement, helpful suggestions, constructive advices, constructive criticisms, and endless patience to take the project and give me direction to complete implementation, write papers, and complete the thesis. The successful completion of the thesis would not been possible without his persistent motivation and guidance. I must also convey my special thanks and gratitude to Dr. Amitabha Chakrabarty, Associate Professor, Department of Computer Science and Engineering for his constant guidance.

I would like to express my gratitude to all of my friends for their helpful suggestions and assistance. I would like to thank my mother for her constant support and encouragement, which have been key in helping me achieve my goals in life. Thank you. For my family, I prefer to do this in Farsi (Dari), my mother tongue:

از طریق دعا های مادرم، تشویق ها، نظر سنجی ها، مهربانی های بی پایان او، نظریات محبت آمیزش به مثل طراوت و تازه گی، روح و روان ام را شجاعت بخشیده است. شادی های های که از طرف خواهرانم برایم مثل باران شبنم صبح گاهی که شاخ و برگم از نسیم روح افزایشان مستفید گردیده است. تا در روشنایی که برایم فراهم ساخته اند در کسب معرفت کوشش به رضای ایزد متعال بی دریغ نمایم. از محربانی بی نحایت همه جانبه و سرشار آنان، برایم توانمندی های غیر انتظار کننده با ارزش ترین تکیهگاه، به کاربرد ذهنی ام در تکمیل پایان نامه تحقیقاتی ام ثمر بخش به اتمام رسید. در سال های دور از وطن در خاک غربت زنده گی ام را رنگ و رونق بخشیدید. نمی توانم با این الفاظ کوچک این همه لطف و مهرابانی ایشان را بیان و تحریر نمایم. یا بار الهی برایشان خوشی، راحتی و زنده گی مملو از برکت، آسایش، عاطفه، صحتمندی مهیا گردان. و من الله توفیق

-* از همه ایشان ممنونم *-

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that are used within the body of the document.

$2CNN$	2-Dimensional Convolutional Neural Network
ACD	Aspect Category Detection
ACP	Aspect Category Polarity
AHP	Analytic Hierarchy Process
$ALQCI$	Afghan Language Query Control Interface
AM	Acoustic Models
$ANLQ$	Arabic Natural Language Query
$ASEM$	Approximate String Evaluation Metric
ATN	Augmented Transition Network
$BERT$	Bidirectional Encoder Representations from Transformers
$BLQPS$	Bengali Language Query Processing System
$BOSW$	Bag-of-Spam-Words
CSR	Continuous Speech Recognition
CTC	Connectionist Temporal Classification
$DBLQ$	Database Language Query
$DBLSTM$	Deep Bidirectional Long Short-Term Memory
DBN	Deep Belief Networks
DNN	Deep Neural Network
GA	Genetic Algorithm

GPSG	Generalized Phrase Structure Grammar
GRU	Gated Recurrent Unit
GUI	Graphical User Interface
HMM	Hidden Markov Model
iPerUDT	Informal Persian Universal Dependency Treebank
IVR	Interactive Voice Response
JAPE	Java Annotation Patterns Engine
KDDA	Kernel Direct Discriminant Analysis
KNN	K-Nearest Neighbor
KWECS	Keywords Extraction and Cosine Similarity
LDA	Linear Discriminant Analysis
LSTM	Long Short-Term Memory
MAS	Microsoft Academic Search
MFCC	Mel Frequency Central Coefficients
MLNN	Multi-Layer Neural Network
MLP	Multi-Layer Perceptrons
MPE	Minimum Phone Error
MSTParser	Maximum Spanning Tree Parser
NBSVM	Naive Bayes - Support Vector Machine
NER	Named Entity Recognition
NLI	Natural language Interface
NLIDB	Natural Language Interface to Database
NLP	Natural Language Processing
NLQ	Natural Language Query
PESD	Persian Emotional Speech Dataset
PNER	Probabilistic Named Entity Recognition

POS	Part of Speech Tagging
QEAD	Query Expansion Assistant database
RNN	Recurrent Neural Network
SA	Sentiment Analysis
SA	Simulated Annealing
SCL	Structural Correspondence Learning
SER	Speech Emotion Recognition
SFC	Securities and Exchange Commission
SFG	Semantic Frame Generator
SID	Semantic Information Dictionary
SQL	Structured Query Language
SVM	Support Vector Machine
TFCMNN	Time Frequency Convolutional Max out Neural Network
TL	Transfer Learning
TPoX	Trade Promotions and Exchange Commission
UD	Universal Dependencies
ULMFiT	Universal Language Model Fine-tuning
VTLN	Vocal Tract Length Normalization
VXML	Voice XML
XML	Extensible Markup Language

List of Publications

1. Sadullah Karimi, Annajiat Alim Rasel, Matin Saad Abdullah (2022). “A review of Persian natural language processing: classical and modern approaches” (submission in progress)

2. Sadullah Karimi, Annajiat Alim Rasel, Matin Saad Abdullah (2022). “Non-English Natural Language Interface to Databases: A Systematic Review” (under review)

3. Sadullah Karimi, Annajiat Alim Rasel, Matin Saad Abdullah (2022). “Natural Language Query and Control Interface for Database Using Afghan Language” (published in INISTA 2022 - The 16th International Conference on INnovations in Intelligent SysTems and Applications),
https://ieeexplore.ieee.org/document/9894168

Copyright warranty: I certify that, when applicable, the Institute of Electrical and Electronics Engineers (IEEE) has granted me permission to reprint the work, text taken directly from the published work, or derivative works for my thesis.

Abstract

Technology adoption is extremely limited in Afghanistan, especially since people have limited access to the Internet, smartphone, and computer due to power limitations and the high cost of the Internet. The people in Afghanistan suffer from high-cost of Internet that is provided by the private sector with very low-speed and quality. Natural Language Processing (NLP) has various applications and improves access to information and systems. To advance as a country, Afghanistan needs to be able to utilize existing databases, datasets, and create new ones and maintain those. Initially, people need a system so they can access the databases providing various guidance with the limited resource that they have access to. Later, they would benefit from higher level access for maintenance and crowdsourced contributions. This work first focus on building a system that Afghanistan people can access database in their native language. Afghan (Dari) language is one of the widely used languages, with up to 110 million speakers worldwide. It is used in countries like Afghanistan, Azerbaijan, Iran, Iraq, Russia, Tajikistan, Turkmenistan, Uzbekistan, etc. The Afghan language lacks resources and requires more qualified lexicon translation. The proposed Afghan Natural Language Interface to Database is based on a natural language query-response model. Afghan language has been used in the model to extract desired data from a database. Retrieving data from a database necessitates knowledge of SQL Query Language or a very well-designed user interface. It is easy for domain experts to retrieve data from databases. However, it is quite challenging for non-expert users to access the database using SQL queries in absence of a proper and friendly user interface. This work overcomes the challenge for those who speak the Afghan Language worldwide to access different databases and datasets. First, we did a survey of current state of Afghan NLP for finding research gaps for future researchers of the Afghan language. We have identified the research gap of NLIDB systems. Second, we surveyed non-English NLIDB systems and conducted a systematic review of the current methods of non-English NLIDB. Then we propose an NLIDB system for Afghan language. Through our system, users in Afghanistan can access the database through feature phone, land phone calls based on an open-source Interactive Voice Response (IVR) system in addition to smartphones and computers. The system can be easily accessed by users without the need for high-speed Internet, sustainable power, computer, and smartphone to access databases. The system is built according to the limited technology situation in Afghanistan. The Afghan Spoken NLIDB build through lexical analysis, semantic analysis, and syntax analysis to respond to the Afghan language natural language query for transforming it into Structured Query Language (SQL).

Keywords: Natural Language Querying; Translating From Afghan to English; Lexical analysis; Syntax analysis; Semantic analysis; Query Generation; Python Library; Data dictionary; Natural language interface to database; NLIDB; non-English NLIDB; Natural language interface; NLI; Natural language user interface; NLUI; Afghan NLP survey; Dari.

Chapter 1

Introduction

Afghanistan is trying to rebuild itself with limited resources almost from scratch. Due to its turbulent history, it has lost many resources including knowledge, skills and know-hows, people, facilities, etc. A small part of the population have access to smartphone, computers, etc. There is irregular supply of electricity, some of which is borrowed from Uzbekistan. Internet speed is not very stable and the cost is very high. As a result, a significant part of the population still uses feature phones. Most of the computing skills are taught by invited temporary faculty members from the neighboring countries. Less than one third of the population can operate computers to be able to utilize different digital services. The portion of population who can develop software and services is negligible. To accelerate its recovery and development, Afghanistan has to utilize all resources and facilities to the fullest extent possible.

1.1 Motivation

Natural Language Processing (NLP) has become an integral part of our day-to-day usage of computers and digital services. It has various applications starting from searching, retrieving information, analyzing, translating, voice-based access and results, etc. Numerous studies have been conducted in NLP to date for resolving various problems. For instance, parts of speech tagging, speech recognition, named entity recognition, grammar and spelling checks, sentiment analysis, dependency, and constituency parsing, natural language database interfaces, document summarization, machine translation, etc. NLP applications have grown over the past several years as computing power has increased as well. To make it easier for people to work, numerous researchers worldwide are developing various NLP applications for their various languages.

One of the most extensively spoken language in the world, with minor variations, is Persian, which is ranked 20th overall. Tajikistan and Uzbekistan both speak Tajiki, Afghanistan speaks Dari, Iran speaks Farsi, etc. Due to the significant similarity of these languages with almost similar vocabulary, grammar, etc., these languages can be thought of as one except some differences in alphabets and pronunciations. Tajikistan and Uzbekistan use Cyrillic alphabets similar to Russia, whereas Iran and Afghanistan uses Arabic alphabets. The area of NLP where computer scientists work on the Persian language is called Persian natural language processing [378].

This research uses the terms “Persian” and “Afghan” interchangeably.

Using NLP, people’s access to digital services and utilization of existing resources can be greatly improved in relatively shorter time without having to wait for much specialized infrastructure and resources, and training for the whole population to be available. The existing population with their existing resources should be able to utilize the systems and access information as much as their devices permit.

The aim is to prepare a model which will change Afghan language sentences to SQL Query. Our main objective is to make it easy and convenient to retrieve data for inexperienced people who do not know how to operate a computer let alone SQL for querying databases, especially for native speakers of Persian in absence of a proper user interface.

1.2 Natural Language Interface (NLI)

A system that users may access using natural language, or our language as humans, is known as a natural language interface. With the use of a speech recognition system, the user may provide input in the form of voice, as well as add text using a keyboard to receive answers to the relevant questions. The user interface for a multi-modal application could need to be in a natural language. An example would be a navigation tool that allows users to interact with a map graphically while also speaking commands and using a pointing device. For instance, a web application may have a chatbot. A person with a physical restriction may be able to utilize voice communication only. The system may have multiple users. For reasons of security and privacy, each user must be individually approved. For example, a person with a physical disability could need to use voice input, text input, single clicking instead of double-clicking, click lock, eye tracking, or a combination of these. A mute person or someone who has trouble speaking may need to use the keyboard. Different output methods or tools are needed by users. A blind user, for instance, would require speech output. The output of text that is visually represented may be required by a user who is deaf or has hearing difficulty. Braille output could be required for a user who is blind and deaf. Users with poor vision or learning difficulties must change the system’s text display settings. Users, especially those with learning or cognitive difficulties, must be able to grasp the terminology used by the system. The system should offer choices for resolving the issue if there is a recognized and controllable limited list of acceptable solutions. Spoken language must be pronounced correctly, especially for users who experience learning or cognitive challenges. Information from text must be presented simultaneously in written and spoken form for some users with learning disabilities. If a time limit is unavoidable, the user needs to be provided the option to extend it before the time runs out, remove it, adjust its duration, or do both. Users also need to be given directions that describe the system’s capabilities [372] [373].

1.3 Natural Language Interface to Database (NLIDB)

Nowadays, databases are the key element for success in any part of life like research, business, government, hotel management, etc. NLIDB act as a significant part for fast and efficient access to data. Accessing databases needs the skill of Structured Query Language (SQL) for retrieving data or an appropriate user interface. SQL is a barrier for inexperienced users whose area of expertise are outside the database domain. It is also difficult for those countries that do not know the English language. To overcome the problem, an Afghan natural language interface to the database is required, as developing an accessible interface is not feasible in the short term. Users can make their query in their native language, and it is easy for data retrieving. For example, in Google, if writing any query in a different language, will retrieve almost exact result in a Graphical User Interface (GUI). The main goal of the natural language interface (NLIDB) to the database is to accept sentences from the user in a native language. After processing, it changes to SQL query. The users enter their spoken using queries in the natural language interface (NLI) search box and receive the exact result as the user wishes. An important part of the database is the Entity Relationship Diagram (ERD) where information about all tables related to a specific database or interconnecting with each other is stored. Tables contain attributes like a student as table name and attribute of students. For example, Student-id, Student-Name, Student Address, Phone-no, etc. So, when a user adds a query in English like **select * from student;** then the data is retrieved from the database. In NLI, users ask in a native language like "show all data" and then after processing it will change to the above query and data is retrieved accordingly. In this research, an interface for database to control and query using Afghan Language is proposed". The system is based on semantic analysis to generate Structured Query Language(SQL) from natural language queries to the database using Afghan language.

1.4 Research Contribution

This research has four key contributions. First, it surveys the current state of Afghan NLP. The resulting survey helped identify different resources that are available for Afghan NLP and multiple areas of research gaps, including the NLIDB. This will help future researchers utilize the resources, easily identify the research gaps, and develop the missing resources. Secondly, it surveys English and non-English NLIDB along with a systematic literature review. It helps to have a critical look at status of NLIDB and help shape design of one. Thirdly, this work demonstrates a proof of concept(POC) design for Afghan NLIDB. Fourthly, it shows future directions that the system can be integrated with Interactive Voice Response (IVR) systems like Asterisk[1] to develop a national helpline for Afghanistan for advices on agriculture, medicine, education, transportation, legal issues, etc.

[1] https://www.asterisk.org/get-started/applications/ivr

1.5 Challenges

The initial challenge was to identify all different resources that could be relevant to Afghan NLP. What data sets are available, what NLP tools are available, all the language variations, dialect etc. After making some progress and identifying the research gap of NLIDB, we have tried Open-Source Software tools accompanying the literature. The majority of these work have been published in conferences, open-access journals, and papers with code available in online search engines, some of which are summarized below. Unfortunately, due to the lack of adequate documentation and troubleshooting that were required, some of their results could neither be reproduced nor made to work for Afghan language.

- Natural language interface to the Relational database which is implemented in Java, using Maven as the project management tool, and JavaFX to design the GUI, the proposal for the system is described in three main steps.

  1. Using a dependency syntax parser for parsing the natural language input into a parse tree.
  2. Mapping the nodes in the parse tree to SQL keywords, Column names, table names, values.
  3. The structure of parse tree should ideally follow the structure of SQL query, like retrieve data according to what the user desired.
  4. Translate the parse tree into an SQL query.

  The mentioned approach works with simple queries, but for complex queries, it shows low accuracy. It may not be adequate to handle natural language problems only by hard-coding procedures. [2]

- Ln2SQL[3] is a NLP tool for querying databases using Natural Language. This approach takes a sentence as an input from the user, and then translates the sentence into a valid SQL statement for retrieving specific data from a database in many languages as trained. The mentioned approach does not need a database connection. The data model learns from parsing of a SQL dump file. It uses tree Tagger to filter input sentence words according to its POS tagging. A mapping between the keywords of input sentences and the keywords of the data model is performed. Tree tagger in ln2SQL is left in choice of an import of personal configuration files for Languages, stop words, synonyms) for more be generic. Ln2SQL will not automatically correct the gender and number problem.

- Natural language interface to database using SIML which is a markup language designed for digital Assistants, NLI for databases, Websites, Games, Chatbots, etc. The approach is implemented in C# language. It utilizes Syn.Bot[4] for creating intelligent Bots or integrate Artificial intelligence and Natural

[2] https://github.com/DukeNLIDB/NLIDB

[3] https://github.com/FerreroJeremy/ln2sql

[4] https://github.com/SynHub/syn-bot-samples

language processing Capabilities to .NET/Xamarin/Mono. It uses SQLite for database. Input sentences from the user changes to corresponding SQL statements for retrieving specific data from database.

Furthermore, due to the lack of sufficient time and resources, the Asterisk IVR[5] could not be integrated to our work.

1.6 Thesis Outline

The remaining part of this thesis report has been organized step by step according to the sequence described below:

- Chapter 2 provides a comprehensive review of the current state of Afghan NLP, its datasets, techniques, and available tools as discovered through studying the literature. It includes a survey of NLIDB which is mostly non-English along with its systematic review.
- Chapter 3 discusses the methodology and components that are part of the proposed system architecture and its design.
- Chapter 4 analyzes and evaluates the result.
- Chapter 5 summarizes the whole thesis and provides further research directions.

[5] https://www.asterisk.org/get-started/applications/ivr/

Chapter 2

Literature Review

There has been a long gap in research in Afghanistan. Many of the researchers have left. There are not many researchers working in Afghan NLP all around the world. Any new researcher willing to contribute to the Afghan NLP may have to face difficulty with access to literature as well as while conducting the fieldwork. This section highlights available datasets, other work in Afghan NLP, NLIDB, and a systematic review of NLIDB.

2.1 Data

2.1.1 Available Persian Corpora

The available sources which include Persian corpora are listed in Table 2.1 that can be used as a secondary dataset or as a reference while creating a new corpus. It includes monolingual, parallel, and multilingual corpora [354].

Table 2.1: Persian NLP corpora statistics

Corpus	Size	Content Type	License	Data Split
English-Persian parallel Corpus	3,500,000 Words	Art, Culture, law, Literature, Medicine others Poetry, Politics Proverb, Religion, Science	ELRA-W0051	Around 100,000 phrases, spread between 50,021 entries
Persian poems corpus	48 documents, each one relating to one poet	Ganjoor.com was used to scrape the data.	Known	48 distinct poets.
Corpus of Conversational Persian Transcripts	20 hours of spontaneously happening Tehrani dialect casual chats	Phone calls, types of relations, and rational goals are recorded from participants.	Abacus Data Network	12 male and 10 female, out of 22 contributors.
SBU-WSD-Corpus	566 verbs, 610 adjectives, 2073 nouns, and 122 adverbs	Arts, Athletics, Science, etc. Persian documents	hrouhizadeh/SBU-WSD-Corpus	5392 content words and 3371 carefully sensed annotated.
Persian (Farsi) Wikipedia Corpus	4,004,765 sentences	Persian Wikipedia	CC0 1.0 Universal Public Domain Dedication	
PerKey Corpus,	553k news articles	From 6 different news websites		
Shereno: A Dataset of Persian Modernist Poetry	4k modernist poems.	Crawled from this website, shereno.com	CC0 1.0 Universal Public Domain Dedication	
MIZAN Corpus	1,021,596 Persian to English distinctive sentences	From web publicly available text.		
ParsTwiNER Corpus	250k tokens	Persian Twitter	MIT license	81.5 F1
Persian 1984 corpus (MULTEXT-East framework)	6,604 sentences	Annotated in the MULTEXT-East framework (Farsi) translation of a section of the novel “1984” (G. Orwell).	ELRA-W0054	100,000 words,13,247 lemmas
Persian Speech Corpus	2.5-hour Single-Speaker	Recorded by male speaker with Tehrani accent in Blubbery studio.	ELRA-S0393	
ShEMO:Persian Speech Emotion Detection Database	Speech data for 3 hours and 25 minutes.	87 aboriginal-Persian rhetoricians which contain (anger, fear, happiness, sadness, and surprise) emotions.		Female = 59.4%, male = 57.6% and 58.2% gender independent.
Corpus of Law, Academic, and News	400 Persian documents included legal, academic, and news genres.	Ten Iranian news outlets between 2010-2020, the civil panel code, the criminal panel code, and the constitution of the Islamic Republic of Iran are included in the legal section.	Abacus Data Network	
CSLU: 22 Languages Corpus	Telephone speech from 21 languages.	Native speakers of Arabic, Cantonese, Czech, Farsi, German, Hindi, Hungarian, Japanese, Korean, Malay, Mandarin, Italian, Polish, Portuguese, Russian, Spanish, Swedish, Swahili, Tamil, and Vietnamese.	10.5683/SP2/PTKKYD	
ARCADE II Evaluation Package	Evaluation of parallel text alignment systems	Evaluation_150325.pdf		
Fas_newscrawl_2015 Corpus	Based on crawled data from 2015	Persian news corpus	Corpus ID=fas_newscrawl_2015	
Pesir_web_2019 Corpus	26,398,268 sentences	Iranian Persian Web text corpus	Corpus ID=pesir_web_2019	

Table 2.1: Persian NLP corpora statistics (continued)

Corpus	Size	Content Type	License	Data Split
CC100 Corpus	Include monolingual data for 100+ languages	By processing January-December 2018 Common crawl snapshots, this was built utilizing the URLs and paragraph indices supplied by the CC-Net repository.	Unknown	
WikiMatrix dataset	135M Parallel sentences in 1620 Languages. 34M are aligned with English	From Wikipedia		
Sentiment Lexicons for 81 Languages	Lexicon languages which include positive and negative sentiment	Created using emotion lexicons in English	GNU General Public License	
Pes_community_2017 Corpus	8,324,536 sentences	Based on content from 2017, the Iranian Persian community corpus was created.	Corpus ID=pes_community_2017	
Persian Lexicon	This is a dictionary of non-inflected versions of words in Persian (Farsi) with over 40,000 entries.	Shargh Newspapers include culture, policy, social, sport, etc.	ELRA-L0087	
Fas_news_2019	462,200 sentences and 11,209,268 symbols	Material from 2019 has been used to create a Persian news corpus.	Corpus ID =fas_news_2019	
Farsdat (Farsi Speech Database)	Include 300 Iranian speakers recording. It Contains 10 Iranian dialect.		ELRA-S0112	
Fas_news_2020	872,034 sentences and 21,769,260 symbols	Based on content from 2020, a Persian news a corpus has been created.	Corpus ID=fas_news_2020	
Fas_newscrawl_2011	7,958,233 sentences and 194,325,652 symbols	Material crawled in 2011 was used to create a Persian news corpus.	Corpus ID= Fas_newscrawl_2011	
Large Farsdat	Including around 73 hours of read Farsi speech	From official texts (newspapers)	ELRA-S0380	
Data from: DAST Dataset	DAST Dataset for Using the DAST Model to Determine Text Semantic Complexity		Attribution 4.0 (CC BY 4.0)	Human Language Understanding, Natural Language Semantics
Bijankhan Corpus	There are over 4300 separate subject categories.	This collection is made up of common texts and everyday news.		Includes 550 part-of-speech tags in Persian
TEP: Tehran English-Persian parallel corpus	Sentence segments: 1.22M tokens: 8.90M	University of Tehran's Natural Language and Text Processing Laboratory		2 languages
CHILDES Farsi Family Corpus	Children's audio	Assembled in the children's separate homes in Tehran	DOI:10.21415/T57K50	
Open Subtitles	Collected number of sentence segments: 3.35G tokens: 22.10G	Is collected from an extensive database of movie and TV subtitles	Opus.nlpl.eu/Open Subtitles 2016.php	62 languages, 1,782 bi texts

2.1.2 Available Persian Datasets

The available Persian datasets are shown in Table 2.2 [354].

Table 2.2: Persian NLP datasets statistics

Task	Dataset	Size	Content Type	License	Data Split
POS	Dari Dataset for Part-of-Speech	12K and hand marked	Azadi Radio and Dari VOA	GNU Lesser General Public License 3.0 (GNU LGPLv3)	
Question Answering system	FarsTail	10,367 samples	Iranian university exam		Training validation and test 70/15/15
	PeCoQ	10,000 complex questions and answer	FarsBase SPARQL		
	PerCQA	989 questions and 21,915 annotated answers	Persian forum Persian Website	Apache-2.0	70% train, 10% Dev., 20% Test
	PersianQuAD	20,000 questions and answers	Persian Wikipedia articles.		82.97% F1
	ParSQuAD	80% and 10% of the total instances, training, and development.	SQuAD 2.0 dataset from translation		SQuAD 2.0 dataset from translation
Information Extraction	Hamshahri	160,000 documents, 65 queries	newspaper articles		
Machine Translation	MIZAN	1,021,596 sentences.	Web publicly available text,		
WSD	SBU-WSD-Corpus	5,892 words	Iranian news websites		
	HmBlogs embedding corpus	6.8 billion tokens	News source, Blogs, Social network		
SA	SentiPers	26,000 sentences	Digikala website		
	Pars-ABSA	600,000 comments	Digikala website		
	Insta-Text	111,000 comments	HALA_KHORSHID Instagram page		
NER	DariNER1		Dari VOA news, Azadi Radio and Kankor	(GNU GPLv3)	
	PEYMA dataset	302,530 tokens	News Websites	NA	84% F1
	ParsTwiNER	250k tokens	Persian Twitter	MIT license	81.5 F1
	ArmanPerso- NER-Corpus	250,015 tokens	Wikipedia, news website	NA	77.45% F1
	ParsNER-Social	205,373 tokens	Telegram channels		89.65% F1
Text summarization	Pasokh Corpus	2,000 man-hours	Top Iranian news agencies		
	PerKey	553k news articles	Agencies and 6 news websites		
Spam and Fake News Detection	Sharif Spam corpus	3,000 classified emails	Personal emails, correspondences, publicly accessible messages, and comments		
Speech Processing and Recognition	ShEMO	3,000 semi-natural utterances	Radio Namayesh Iran		Female=59.4%, Male=57.6%
	MirasVoice	50 individuals speaking 2 languages on 4 distinct texts	Conference room at Miras Technologies International		
	DeepMine	1,850 speakers and 540 thousand recordings	Android application.		
Parsing	Uppsala Persian Corpus	2,703,257 words with morphosyntactic and partially semantic annotations'	newspaper topics like technology, fiction, culture, and art		
Plagiarism detection	PerPaDa	2,446 instances of paraphrasing.	Hamtajoo		

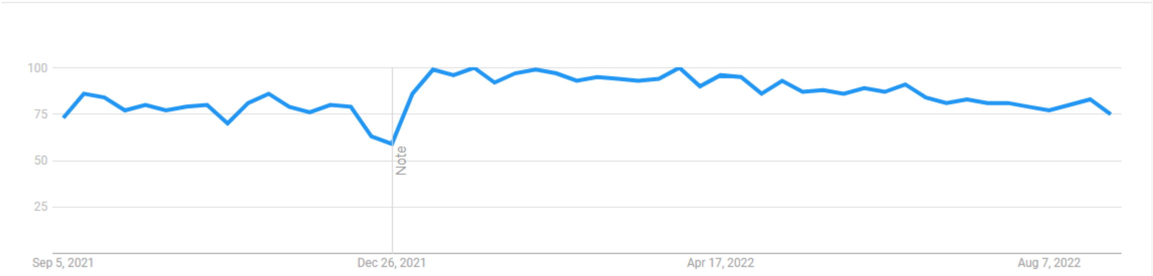


Figure 2.1: Interest over time in “Natural Language Processing” Google Trends statistics

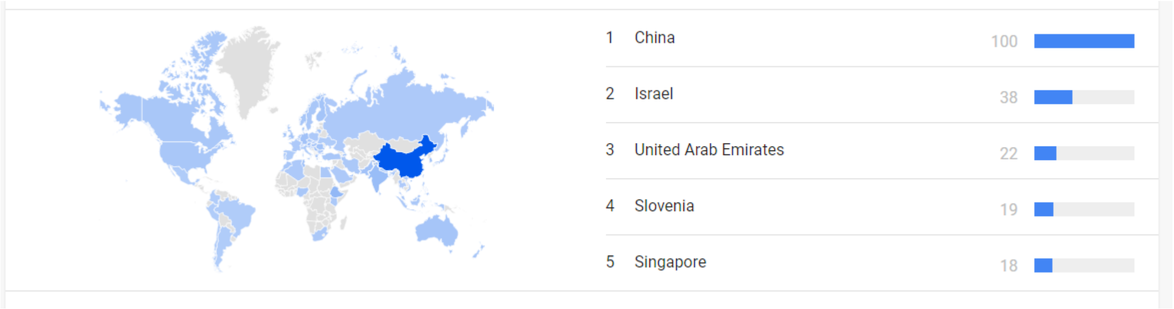


Figure 2.2: The interest of the countries in “Natural Language Processing” Google Trends statistics

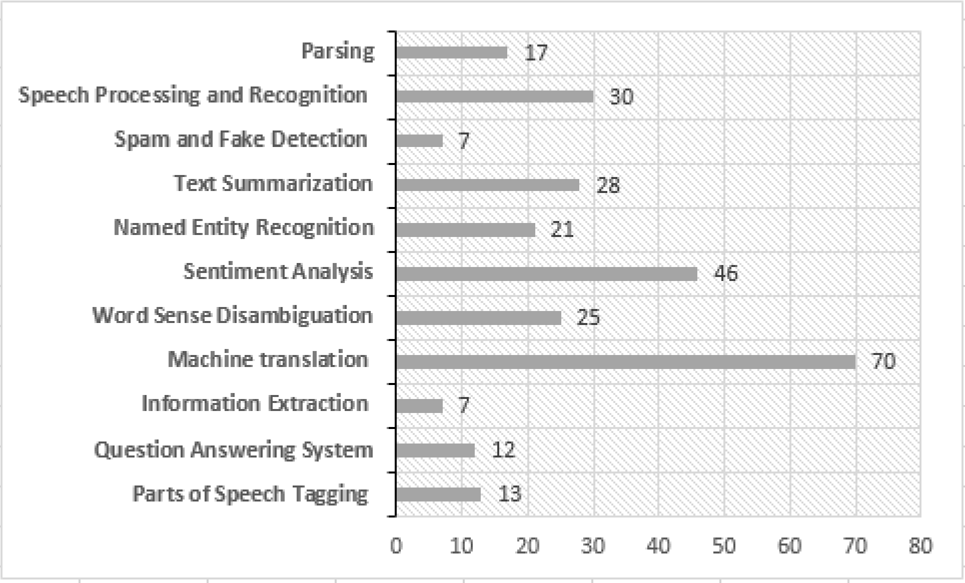


Figure 2.3: Replicated papers in the category

2.2 Persian Natural language Processing

2.2.1 Persian Text Processing Toolkits

Sakar Mohtaj et al. [203] created a text processing toolkit for the Persian language that includes space correction, tokenization, stemming, POS tagging, and shallow parsing. As a downstream action, the extrinsic evaluation of the recommended toolset made use of a Persian plagiarism detection system. The results show that the Parsivar toolkit outperforms Persian preprocessing toolkits in terms of F1 by about 8%. Additional work may be done in future to improve the effectiveness of the shallow parser utilizing different techniques.

Romina Etezadi et al. [350] developed the DadmaTools, a Persian text processing toolkit built on the spaCy Python library. The toolbox may be used to handle text in numerous ways, including normalization, tokenization, lemmatization, part-of-speech, constituency, dependency, chunking, and Ezafe[1] identification. The toolkit also supports word embeddings and datasets. The author plans to include sophisticated NLP tasks like sentiment analysis, entailment, and summarization to the supported tasks in the next work. The author also wants to allow users to add new datasets and models to the toolkit. The source code is available to Persian academic researchers without charges.

Peng Qi et al. [282] completed the 66 human language-supporting Stanza tool set for text analysis, which is frequently utilized by Persian scholars. Stanza's fully neural pipeline for text analysis comprises named entity recognition, dependency parsing, part-of-speech and morphological feature tagging, lemmatization, and part-of-speech and multi-word token expansion, in contrast to presently popular toolkits. Using a total of 112 datasets, including the Universal Dependencies treebank[2],[3] and extra multilingual corpora, the author demonstrates that the same neural architecture generalizes well and provides competitive performance in all languages. For 66 distinct languages, researchers have access to pretrained models, documentation, and source code for Python.

Weston Feely et al. [111] constructed the Farsi NLP tools, text pre-processing, dependency parsing, and part-of-speech tagging. The author has enhanced the available Farsi resources by creating the additional text processing tools, text normalizer, and verbal morphology tokenizer required to utilize these tools to create accurate parses on new Farsi texts. All of the tools, in addition to the previously free ones were used to build this Farsi front-end, are freely available online.

Dat Quoc Nguyen et al. [204] proposed a novel neural network model for integrated graph-based dependency parsing and POS labeling that is also applicable to the Persian language. On the benchmark English WSJ Penn treebank, the model gets strong parsing scores with UAS[4] at 94.51 percent and LAS[5] at 92.87 percent, as well as a cutting-edge POS tagging accuracy of 97.97 percent.

Mehrdad Farahani et al. created [312] ParsBERT, a unique model that, despite being lighter than multilingual BERT, achieves state-of-the-art results in downstream applications including sentiment analysis, text categorization, and recognition of named entities. When compared to other Persian NER competitor models, ParsBERT outperforms all prior studies in terms of F1 score, achieving scores of 93 percent and 98 percent for the PEYMA and ARMAN datasets, respectively. Additionally, ParsBERT surpassed DeepSentiPers in the Sentiment Analysis job by achieving F1 scores of up to 92 percent and 71 percent for binary and multi-label circumstances, respectively, on the SentiPers dataset. Compared to multilingual BERT and other recommen-

[1] https://sites.la.utexas.edu/persian_online_resources/language-specific-grammar/ezfe/

[2] https://faculty.washington.edu/fxia/lsa2011/slides/intro_to_treebanks.pdf

[3] http://faculty.washington.edu/fxia/lsa2011/slides/create_treebank.pdf

[4] https://www.mff.cuni.cz/veda/konference/wds/proc/pdf11/WDS11_123_i3_Green.pdf

[5] https://aclanthology.org/W17-0411.pdf

dation networks, ParsBERT routinely outperforms those. ParsBERT embracing public usage and sets a new standard for numerous Persian NLP use cases.

Mohammad Abdous et al. [348] worked on semantic text similarity between Persian and English sentences based on deep learning. Applications that take advantage of the semantic similarity of texts include machine translation, fraud detection, question-and-answer systems, information retrieval, and others. Comparing two textual passages, phrases, or documents that are available in monolingual and multilingual versions is necessary to determine semantic similarity. This paper presents the first cross-lingual model of semantic similarity for Persian-English words using the parallel corpus. They tested their model and contrasted it with Multilingual BERT models. The findings show that using parallel corpora can improve the quality of sentence embedding in two different languages. The cosine similarity between the sentence vector of multilingual BERT and the recommended approach increased the Pearson correlation threshold from 65 to 73.77 percent. The findings show that the suggested method outperforms the multilingual BERT on the Arabic-English language pair.

Parsa Kavehzadeh et al. [357] suggest a Transformer-Based method for Persian Text Chunking. The models were mostly built using the newest transformer-based contextualized models, namely BERT and XLM-RoBERTa. Following the transformer-based models, the performance of the model in predicting the chunk labels is enhanced using a conditional random field (CRF), a mix of bidirectional long short-term memory (BiLSTM) and CRF, and a fundamental dense layer. They provided a brand-new dataset for chunking Persian noun phrases derived from annotated Persian news articles. Their research demonstrates that XLM-RoBERTa outperforms all other designs when evaluated against the specified dataset. According to the gathered data, using a single CRF layer results in better outcomes compared to using a thick layer or even combining CRF with BiLSTM.

ExaPPC from Reyhaneh Sadeghi et al. [359] is perhaps the first significant dataset of Persian phrases used for paraphrase identification. With 1.3 million non-paraphrase labels and 1 million paraphrase labels, the corpus has 2.3 million tagged sentence pairs. Using techniques like subtitle alignment, translating an already-existing parallel English-Persian corpus, and similarity corpus on English tweets, this corpus was built both manually and automatically. In addition to growing the corpus, two Persian native speakers labeled potential sentence pairs that were extracted from tweets. This corpus is superior to existing because of the quantity of pairs of sentences, the range in sentence length, and the literary variety, including formal and conversational phrases. Results from the test corpus demonstrate that ExaPPC provides 94% accuracy on the paraphrase identification task [350]. The corpus may be seen in the public domain at [6]

[6] https://github.com/exaco/exappc

Table 2.3: NLP toolkits for Persian and their features

Toolkits	Normalizer	Lemma	POS	Dependency	Constituency	Chunker	Ezafe
Stanza[7]	No	Yes	Yes	Yes	No	No	No
spaCy[8]	No	Yes	Yes	Yes	No	No	No
Hazm[9]	Yes	Yes	Yes	Yes	No	Yes	No
farsiNLPTools[10]	No	No	Yes	Yes	No	No	No
Perstem[11]	No	No	Yes	No	No	No	No
persianp Toolbox[12]	No	No	Yes	No	No	No	No
UM-wtlab pos tagger [13]	No	No	Yes	No	No	No	No
RDRPOSTagger[14]	No	No	Yes	No	No	No	No
jPTDP [15]	No	No	Yes	Yes	No	No	No
Parsivar[16]	Yes	No	Yes	Yes	No	Yes	No
text mining[17]	Yes	Yes	Yes	No	No	No	No
DadmaTools[18]	Yes	Yes	Yes	Yes	Yes	Yes	Yes

2.2.2 Parts of Speech Tagging

Mahdi Mohseni et al. [42] provided a method for a Persian language part-of-speech tagging (POS) system built on morphological analysis. It pertains to the Peykare, a sizable corpus (or Textual Corpus for the Persian Language). It is a well-known corpus in Persian. As may be seen here, the Peykare corpus is split into two pieces.

Unannotated Parts: It includes a conversational text that was incorporated from Persian storybooks, interviews, and plays. There are 90 single tags in the corpus, while the remaining 16 tags include crucial categories including adjectives, adverbs, verbs, and nouns. There is a hierarchical structure to the corpus. Many of the tags were removed early on by the author and were not POS tags. The single tags for place (LOC), direction (DIR), day (DAY), season (SES), surname (SURN), month (MON), and time are liberated from the noun class. These tags are nouns and adverbs in particular. The terms' location (LOC), time (TIME), example (EXM), ordinal (ORD), negative (NEGG), and repetition (REPT) are eliminated from the category of adverbs, and when tagging is complete, adding tags to words is simple. There are now just 471 distinct tags left in the corpus. The formal and conversational forms of morphemes are discussed separately in the next step, since they are not interchangeable. The number of tags decreased from 471 to 105 tags in the following stage, therefore all words with the same lemma are combined and not gathered separately. The next stage is to develop a lexicon

[7] https://stanfordnlp.github.io/stanza/

[8] https://github.com/explosion/spaCy

[9] https://github.com/sobhe/hazm

[10] https://github.com/wfeely/farsiNLPTools

[11] https://github.com/jonsafari/perstem

[12] https://github.com/mohsenim/persianp

[13] https://github.com/mhbashari/awesome-persian-nlp-ir/blob/master/sections/tools.md

[14] https://github.com/datquocnguyen/RDRPOSTagger

[15] https://github.com/datquocnguyen/jPTDP

[16] https://github.com/ICTRC/Parsivar

[17] https://www.text-mining.ir/

[18] https://github.com/dadmatech/dadmatools

for each new term that is added to the dictionary, complete with its morphological analysis. After the lexicon is created, the dictionary may search for any term to retrieve its morphological analysis. Before tagging a new document, each word is replaced with its most common morphological analysis word that is already registered in the wordbook in this stage. A test indicated a 5% mistake rate in morphological word analysis. In the last stage, the tagger is applied to the words that were inspected. The goal of the current work is to expand the massive Persian Peykare corpus to develop a system for classifying words using morphological analysis. The anticipated likelihood of POS tagging for unidentified words is 39% for common nouns, 18% for proper nouns, 25% for simple adjectives, 2% for verbs, 11% for residuals, and 5% for others. The system's accuracy cannot exceed 53% due to the system's large (105) number of unknown word number tags.

Mahdi Mohseni et al. [24] suggested a Markov model-based method for labeling Persian parts of speech. It is applied to the ”متن زبان فارسی”, “Persian language text”. The corpus contains over 10 million words and 7.5 million annotated tokens. Numerous works, such as those on politics, text genres, society, culture, economics, art, sport, and religion are included in the corpus. The remaining 25 major categories and 168 tags in the corpus are for the Farsi morphology. The author used 25 main tags for simulations. The system applied 10-fold cross-validation and approximated the probability for unknown words. 60% for nouns, 20% for adjectives, 10% for verbs, 2% for adverbs, and 8% for others. Verb+ (conjunction, preposition, ‘.’,‘?’, ‘:’or ‘,’), these concepts are utilized for catching sentence boundary. The system acquired 97.5% accuracy for well-known words, 68.0% for unknown words, and overall 96.9% accuracy of this work. On the referenced corpus and upper class of the tag set, and quality of the Markov Model of POS tagging for the Persian language.

Abbas Koochari et al. [276] propose a neural network-based, long short-term memory-based Persian part-of-speech tagging approach. It is used with the Bijankhan corpus, which has over 2 million words, 600,000 labeled words, and comprises a significant quantity of Persian literature. There are 550 parts of speech tags in the corpus. The corpus includes theme materials on politics and history, in addition to more than 4,300 tags. Following processing and normalization are used to get the corpus ready and transform the crucial data for the neural network training phase.

1. Finding out the required number of tags.
2. Word standardization and detachments.
3. Probabilities and statistics extraction.
4. Ambiguous word problem-solving.
5. Problem abstraction to a time series.

Longer-term memory that has been trained and tested is combined with probabilistic and statistical characteristics and information utilized with LSTM neural networks. Prefix, suffix, stemming, and evaluation of bidirectional words were taken for LSTM neural networks for increasing dependence length of preceding words.

In comparison with several cutting-edge methodologies, the present system obtains the accuracy of 98.1 percent. The LSTM performed well compared to other structures when dealing with suffixes, prefixes, five prior words, stemming, and human mistake correction; nevertheless, one drawback of this approach is that it takes a lot of time.

Pejman Golami-Dastgerdi et al. [314] came up with the sequence graph approach, which is used to describe a technique for tagging Persian elements of speech. They allocated well-known and unknown terms that are included in the text by offering a graph-based methodology for correcting those using the Maximum Likelihood Estimation method. In this study, graph sequence-based techniques works by using a Maximum Likelihood Estimation (MLE) method to tag input text at first. The second stage includes defining the sentence boundaries. Sentences are forwarded to the third-stage, detection of individual sentences as graph paths in the speech graph. The selection of pathways on the graph using statistical methods is one of the feasible options. Sentences are marked in the final step with the pattern that was pulled from the graph. The final accuracy is 96.78% the consequence is standard compared to the MLE method.

Sara Besharati et al. [301] demonstrated a Persian part-of-speech tagging system based on statistics and deep learning methods. The suggested approach was applied to two different Persian corpora, the first of which is the Persica corpus and is utilized for text mining and natural language processing. Persian News writings are among the several items in the corpus that total more than a million words. In this study, word embedding is done using the Persica corpus. The Bijankhan corpus is another corpus used in this study. It has ten million words and more than 4,200 subjects. This corpus's first version had 607 different tag types. It is difficult to predict words that are both in- and out-of-vocabulary (IV) in part-of-speech tagging. Both the Hidden Markov Model (HMM) and the neural model are employed in this study, although the neural model presents improved performance for foretelling and tagging out-of-vocabulary words (OOV). The major goal of the hybrid model in this study is to get around the difficulties posed by the enormous number of OOV terms. It can increase the precision of HMM and Long Short-Term Memory (LSTM) models. The author will use a much larger dataset to construct the system in subsequent research. They will benefit from using recurrent neural network models if they encounter a larger dataset or body of data.

Morteza Okhovvat et al. [55] based on the Markov model, provide homogeneous and heterogeneous parts of speech tagging for the Persian corpus. The primary aspect of Persian morphology for defining the sentence boundary between morphemes is presented and developed in the suggested strategy for achieving this goal. 98.1-percent accuracy was achieved in the experiment on a Persian corpus.

Mohammad Javed Rezai et al. [183] Offer Farsi Tag, a potent tagger system for Persian, developed after doing several assessment and effectiveness tests. The system demonstrates that, with a 1.4% mistake rate for known and unknown terms, the existing tagger can successfully handle the tagging of limitless Persian text. Authors avoid the genotype selection of each uncertain phrase that the underlying algorithm handles. The morphological analyzer is introduced to improve speed and quicken its tagging procedure. In this study, a bilingual parallel

English-Persian corpus with part-of-speech tags is published. This corpus has numerous uses in NLP tasks, the most significant of which is statistical machine translation. The algorithm achieves 98.6% total accuracy, which is encouraging for Persian NLP.

Zeinab Mirzanezhad et al. [152] provide automated processing and analysis of Persian morpho-syntax. The present method makes use of the Bijankhan corpus, a Persian-tagged corpus. The author improved part of speech-tagged corpus and built tools for preprocessing Persian text like text normalization, tokenization, POS, and Chunking. The system's evaluation reveals a 16-percent capacity to identify unidentified words using the morphological analyzer. Up to 70% better results are obtained when the "Singular Noun" tag is used when words are not detected.

Fahimeh Raja et al. [21] evaluate the statistical part-of-speech tagger Tri-grams'n'Tags (TnT), which uses the Markov model and linear smoothing on the Persian language. As long as the test and training sets are measured as 10 and 90 percent of the corpus independently, experiments using TnT tagger for Persian text demonstrate the maximum overall accuracy of 96.94 percent. The results indicate that recognized words are more accurate than unfamiliar words by approximately 18%. The size of the test and training sets affects TnT's accuracy. The gap between the small size training text result and the top overall result is 0.50.

Mitra Mohtarami et al. [25] give a review of maximum probability techniques for labeling Persian segments of speech. According to the results, the MLE-N_SING model's overall exact tagging of 95.29 presentations was demonstrated. Using new heuristic guidelines that aim to be virtually as accurate as the Persian part-of-speech tagger. The accuracy of the system is influenced by the utilization of large, intelligent corpora. Lastly, a collection of post-processing heuristic guidelines that raise the MLE tagger's caliber. The accuracy is raised to 1.5 for the MLE-DEFAULT model and 0.86 for the MLE-N_SING. The current approach uses the Bijankhan Persian tagged corpus for tagging.

Mojgan Seraji [61] introduces HunPoS, a statistical part-of-speech tagger that was developed using the Bijankhan corpus. HunPoS has the highest results for Persian part-of-speech tagging yet published, with trials showing an overall accuracy of 96.9 percent.

Ahmad A.Kardan et al. [116] give a strategy for improving the accuracy of part-of-speech tagging for the Persian language. The sliding windows in the initial stage have sentence tags. The dataset is used to extract each word's orthographic properties, affixes, and other heuristic features. To tag POS, machine learning classifiers and maximum entropy are used. The method was tested using the well-known Persian POS-tagged corpus Bijankhan. The findings demonstrate that the proposed strategy outperforms previous approaches in the aforementioned dataset in terms of tagging precision, with a 97.53-percent improvement and an F1 measure of 92.67 percent.

Zahra Hosseini Pozveh et al. [150] propose a technique for classifying Persian linguistic components using artificial neural networks. The suggested approach is designed to get around word ambiguity, named entity recognition, and composite verb recognition. The advantage of this technique is that it requires fewer iterations to identify POS tags on words. In comparison to stochastic and other optimization models, the outcome demonstrates that artificial neural network (ANN) accuracy is high. ANN is basically equivalent to other algorithms for well-known word annotation. The accuracy rate significantly increases when Named-Entity Recognition (NER) is used with the Elman neural network (more than 4 percent). Clustering rises more than 5% to POS tagging because certain semantic components are included in the model. While ANN produces answers that are almost identical 50% of the time, other ways employ semantic analysis to improve the Ezafe identification process [354].

Table 2.4: Persian Parts of Speech Tagging (POS)

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mahdi Mohseni et al. [42]	Peykare, a Persian large-scale corpus, accomplishes a total accuracy of 53-percent	Removing disjointed tags and many of them are not POS tags. Morphological analysis
Mahdi, Mohseni et al. [24]	Persian language text Corpus. 96.9-percent total accuracy.	Hidden Markov Model (HMM)
Abbas Koochari et al. [276]	Bijankhan corpus, acquire 98.1-percent total accuracy.	Preprocessing: suffix, prefix, 5 previous word, stemming and manual mistake correction, long, short-term memory neural network.
Pejman Golami-Dastgerdi et al. [314]	Bijankhan corpus, acquire total accuracy of 96.78-percent	Preprocessing done to raise the accuracy of known words, sequence graph method,
Sara Besharati et al. [301]	Persica corpus utilized for word embedding, Bijankhan corpus applied to raise the accuracy of LSTM, HMM, and also cut down the difficulty of OOV words	Statistical technique and deep learning
Morteza Okhovvat et al. [55]	Persian corpus, outcome shows 98.1-percent accuracy	Heterogeneous and homogeneous POS for Persian established on Markov model.
Mohammad Javed Rezai et al. [183]	FarsiTag, 1.4-percent mistake rate in tagging limitless Persian text. With 98.6-percent total precision.	Create FarsiTag for Persian POS tagging by increasing morphological analyzer.
Zeinab Mirzanezhad et al. [152]	Bijankhan Corpus outcome uncovered 16-percent ability for discovering unknown words.	For POS tagged corpus, text normalization, tokenization, POS and chunker for Persian, using morphological analyzer, analysis of automatic morphosyntactic Persian text.
Fahimeh Raja et al. [21]	Outcomes of known words are 18-percent larger than unknown words, with a total accuracy of 96.94-percent.	Measure TNT tagger on Persian text with exploiting the linear smoothing and Markov model
Mitra Mohtarami et al. [25]	Bijankhan corpus, MLE-N_SING, MLW model has total fine tagging approximately 95.29-percent	Post-processing heuristic rule for raising the quality of Persian POS tagger. With the maximum likelihood method assessed the tagger.
Mojgan Seraji [61]	HunPoS renders total accuracy of 96.9-percent, which is the highest outcome for Persian POS tagging ever publicized	Statistical POS tagger HunPoS which trains on Persian corpus.
Ahmad A.Kardan et al. [116]	Bijankhan, POS corpus system raises the quality of tagging precision with 97.53-percent and F1 score of 92.67% advance than some other approach in the mentioned dataset.	Maximum entropy and machine learning classifiers are used to POS tag, to raise the quality of Persian POS tag.
Zahra Hosseini Pozveh et al. [150]	System can find part of the speech tag of a word in less time. The system shows high accuracy compared to stochastic and other optimization models	The system utilized ANN to get rid of composite verbs, named entity recognition, and ambiguity of word recognition in Persian Part Of Speech tagging.

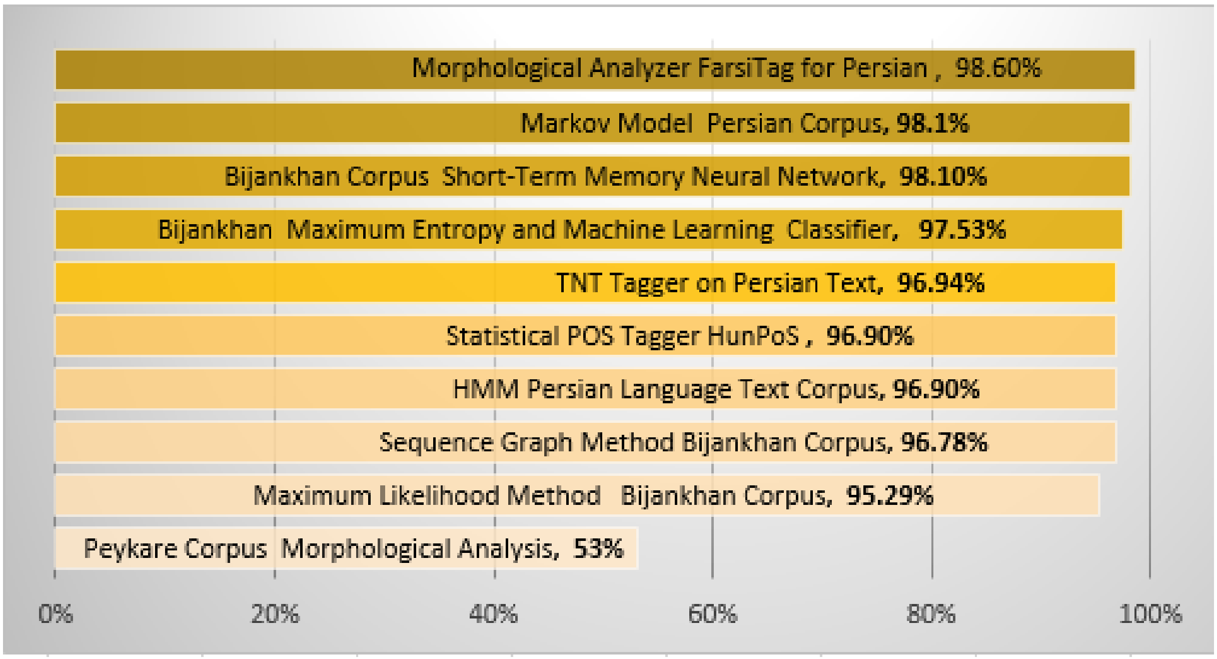


Figure 2.4: Part of Speech Tagging methods performance in different corpora

2.2.3 Question Answering System

Hadi Veisi et al. [292] provide a methodology based on dictionaries, natural language processing, and rule-based procedures for the Persian question answering system for the medical system. The method uses a dataset of illnesses and medications that the author of this work has organized. The present collection consists of 1,767 documents with semi-structured information regarding medications and illnesses. When a user adds a query, the system is activated. The query is standardized such that the named entity dictionary can extract the matching names of the medicine and illness. After identifying the name, the question is sent to the question-classifier component for phrase extraction and question meaning extraction. The retrieved keywords and phrases are mapped to dictionaries at the very end by utilizing the dictionary keywords. The question is transferred to the n-gram module to extract the keyword from the keyword dictionary if the question classifier is unable to extract any keywords at that point. The question topic was extracted by the author using a pattern matching component that has a set of rules. When the question classifier and n-gram failed to extract the question's topic, the pattern matching system succeeded. After stopping at stop words, the question processing module utilizes an advanced tokenizer to lemmatize the word and replace it with the appropriate lemma. The extracted keyword is then transferred to document retrieval for the response extraction module after successfully completing the above procedure, and the answer is then selected from the recovered document. The document is transformed into a structured form to allow for more precise answer extraction. The Apache Lucene library[19], an effective and potent text processing tool, is utilized by the author to search and index the document in the document retrieval module. 500 sample questions were answered with an overall accuracy rate of 83.6% by the method.

[19] https://lucene.apache.org/

Omid Moradian Nasab et al. [371] offered a method for the Persian question-and-answer system that was based on a unique domain. A system similar to the Google search engine was created by the author. Following data retrieval from a particular domain dataset, the system accepts input in natural language. The procedure is shown below. Users begin by asking queries on the inquiry interface.

- **Normalizer:** This phase involves removing punctuation, language characters, white space, and some unidentified characters. Every character membership is replaced and verified by a corresponding previously contracted one by a normalizer. It is used in example words to get rid of mistakes that occur when Persian alphabets are duplicated in Unicode charts.

- **Spell checker:** These part functions before the user query processing begins. In the user queries, it looks for misspelled words in the dictation. It notifies the user if any mistakes are discovered. In unified Persian, incorrect words are covered by it.

- **Tokenizer:** It is used to separate words from sentences. For example, various prefixes and suffixes in Persian grammar may or may not attach to the word; this component discovers and identifies words. Those were then divided so that the link grammar parsers could tag them more accurately and effectively.

- **NLP Parser:** the Part of Speech tagger in this component works by connecting the grammar parser to dependency-like grammar. For identifying verbs, subject-object connections, and other grammatical relationships inside a phrase.

- **Concept extractor:** this part is responsible for determining the semantic relationship in the user question. This component receives the output from the link grammar; the question is divided into the predicate, subject, and object portions. The query word is expected to be in one of these parts.

- **Query Reformulation:** The goal of this component is to translate these synonyms into machine-readable code and create the idea of ontology. For example, the character "ls-in" is rephrased to "located in" when used in an ontology. The strategy is built on WordNet innovation. Punctuation marks are removed from this component during processing, words are stemmed, and words that make sense in terms of the ontology idea may be used in place of synonyms.

- **WordNet:** The author of this work created a customized word net based on an XML format for the present system that contains terms relevant to a certain topic.

- **Ontology:** The author made use of a custom ontology that had concepts for unique domains. Ontology is the formal representation of knowledge in a particular domain and a model of that domain built from ideas related to a certain domain. The OWL language[20] is used to construct and express the existing ontology on the Stanford Protégé stage.

[20] https://www.w3.org/OWL/

- **Knowledge base:** It was produced using the specified ontology. Each phrase that expresses knowledge in natural language is built from two relationship objects called object-properties.

- **Reasoner:** It is software that draws logical conclusions from a set of stated facts. For the final version of the OWL reasoner that includes the Apache Jena Framework[21], which is utilized in Java and is an open-source semantic web framework, the author employed several modern semantic reasoners including FaCT++, HermiT, and Pellet [22].

- **Knowledge extractor:** It requires a triple semantic return. Return that if it is accurate or if any of the three components of this triple are missing. This part was created in Java using the Jena web framework.

Negin Abadani et al. [294] built a ParSQuAD dataset for the Persian question-answering system. Using the Google neural machine translation (NMT) API, the aforementioned dataset converts the SQuAD 2.0 dataset to the Persian language (API). The author stumbled into one issue while translating. The referenced API is used to change Hamza (ء) with English words. The Arabic alphabet is used to write Hamza, since the Persian language uses an Arabic-based character. The term with the Hamza letter cannot be correctly translated using the Google Translate API. The author initially constructed an automatically generated dataset and then a manually generated dataset as a result of the aforementioned problem. When evaluating the manually created dataset, the author of this research encountered a concern where certain spans were not corrected due to punctuation. The second issue was that the Google-translated API failed to translate the letter from Hamza in the context of the paragraph, inside both single (‘’) and double (“”). This challenge led the author to attempt manual corrections. Instead of addressing the problem in the second, automated dataset, he completely deleted the text and any additional passages that it was paired with. The author examined current datasets using ParsBERT, ALBERT, and multilingual BERT (mBERT), and was able to get an F1 score of 56.66% with an original ratio of 52.86%. For the first version system, an F1 score of 70.84% with an original ratio of 67.73% for the second version. A 56.66% F1 score was attained by the mBERT model.

Ali Eshani et al. [308] gave a rhetorical structural theory-based approach to the Persian question-answering system. The algorithm seeks to extract “why” or “how” related queries in Persian. Rhetorical structure theory, commonly known as a text structure recognition method, is used for text and data analysis. 70 “why” questions and 20 “how” questions were used to assess the performance of this system addition. The current study was constructed using a methodology with a focus on design science research (DSR). The following steps showed the system architecture. The next package employs Tokenizer and POS tagger, while text parsing uses “Discourse Markers” to detect relationships between phrases. Pattern Recognizer is used to create inter-sentence linkages in the text. The author referred to the “Stemmer” in the third package using “answer finders” and “getting keyword”. The total accuracy of the method is 68%.

Romina Etezadi et al. [309] afford a knowledge-based strategy for answering questions in Persian that can deal

[21] https://jena.apache.org/

[22] https://its-wiki.no/wiki/Comparison_of_Pellets,_FaCT_and_HermiT

with challenging ones. The system is developed using PeCoQ, a recently produced dataset for answering complicated questions in Persian. PeCoQ has more than 10,000 complex questions (1,000 dev, 1,000 test, and 8,000 train[23]), and it is generated automatically using data from FarsBase. The author of this paper has experience with multiple-hop and multiple-constraint questions. The author utilized dependency trees and Multilingual-BERT for the current piece. The semantic matching approach eliminates links with a score below a threshold and reduces the search space. To improve the PeCoQ dataset, the author of this research advises concentrating on adding more complicated inquiries as well as enhancing semantic matching and number hops in the supplied query. The program succeeded in achieving an overall accuracy of 62.75%.

Nasim Tohidi et al. [290] provide a method for improving the caliber of Persian-language web-based question-answering. The suggested solution considered a Persian multiple-aim question answering system that would fetch the page from the excellent search engine that would include nearly all likely responses. Lexical aspect, contextual aspect, and syntactic aspect are all considered when looking at the text from all angles. The author employed NSGA-II as a Multi-Aim developmental algorithm (MOEA) for an experiment including the present strategy, as well as a Rasekhoon question-answering dataset that had 927 Persian words and web data. Comparing the outcome to the current system, the result was efficient.

Daniel Khashabi et al. [322] developed ParsiNLU, the first Persian language standard with a range of language comprehension, duty comprehension, reading, and entailment textual. The current dataset was collected using a variety of techniques, mostly hand annotation by native speakers. This leads to the creation of almost 14.5k new instances across 6 separate NLU tasks. Modern multilingual and monolingual pre-trained language models were utilized in the experiment, which generated a qualitative grasp of our ability to execute the issues of Persian natural language comprehension in comparison to human quality.

Yasaman Boreshban et al. [194] created the Persian question and answer database Rasayel&Massayel. For each question in the existing corpus, there are 2,118 false questions and 2,051 genuine questions. The question's content, format, and difficulty from the perspectives of the asker and the respondent. Look for both fine- and coarse-grained response kinds. The precise response, along with the relevant page and many answer paragraphs, are marked. For NLP areas, the existing corpus is beneficial. This includes finding answers, classifying questions, retrieving information, and answering questions. Researcher access to the dataset is unrestricted. For the matching rank, the experiment of the present corpus shows 82.29-percent accuracy and 56.73 percent mean.

Ehsan Sherkat et al. [124] provide a closed domain question classification strategy for the Persian language based on rule-based and machine learning methods. The suggested technique is using the online automatic question-answering technology known as quranjooy. The outcome was what the method called for.

Naghme Jamali et al. [318] introduced PerCQA, the first Persian-language dataset for addressing community questions. The current dataset, which is available to academics in the public, has 989 valid questions, and, 21,915 valid responses. By applying both multilingual and monolingual pre-trained language models, the au-

[23] http://cs230.stanford.edu/blog/split/

thors additionally provide rules for the PerCQA obligation of response selection.

Ali Mollaei et al. [79] suggested a strategy for employing a Conditional Random Field (CRF)[24] in the categorization of Persian questions. It is a machine learning model, and several characteristics have been tested to see how accurate it is. Based on the type of response to each individual inquiry, sentences were classified in the current study into two levels of fine and coarse classes. Question classifier CRF is trained when features are highlighted and sliding windows are placed on the CRF model. The classifier predicts labels for each of the tokens under consideration. To give each question a distinctive name, they use our categorized question on a majority vote. Different feature impacts on the system's overall accuracy were evaluated. The end result met the needs of the existing question classifier and was effective.

Mohammad Razzaghnoori et al. [210] provide a method for categorizing Persian questions based on Word2Vec that uses a vector format to hold numerous precise syntactic and semantic links. The author discovered that TF-IDF weighting had a significant impact on improving accuracy. To get a successful output, significant attention should be paid to the TF-IDF component. The outcome was satisfactory; these questions were prepared to be challenging for humans as well.

Zeinab Borhanifard et al. [261] introduce a natural language understanding-based entity recognizer and ParsBERT chatbot system that is tailored specifically for Persian online shopping. An Iranian e-commerce website called Digikala, which has 3,600 conversations about items for online buying, is used to collect a dataset. The cold start problem in a low-resource language can be resolved in a task-oriented conversation system using the suggested dataset generation techniques in numerous academic fields. The system's final performance results in a precision of 86/71, recall of 52/5, and an F1 score of 65/20 in a 3,000 dialog trains size. F1 scores increased by 5.4% when the NLU model was applied to the combined dataset. Jamshid Mozafari et al. [358] the first comprehensive Persian native Answer Selection (AS) Dataset, known as PASD. To show the efficacy of PASD, authors trained state-of-the-art QA systems to develop PerAnSel, a novel deep neural network-based system for Persian question answering. Because Persian is a free word-order language, the inventor of PerAnSel parallelized a sequential technique and a transformer-based approach to handle various word orders in that language. Then, they evaluate PerAnSel using three datasets: PASD, PerCQA, and WikiFA. According to the experimental results, the Persian datasets surpass state-of-the-art answer selection methods in terms of MRR by 10.66% on the PASD dataset, to 8.42% on the PerCQA dataset, and 3.08% on the WikiFA dataset [354].

[24] https://people.cs.umass.edu/ mccallum/papers/crf-tutorial.pdf

Table 2.5: Persian Question Answering System

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Hadi Veisi et al. [292]	Diseases and drugs dataset which is arranged by author, 83.6-percent accuracy for 500 sample questions.	Advanced tokenizer for lemmatization and renewing the word with a related lemma, rule-based, natural language processing, and dictionary-based methods.
Omid Moradian Nasab et al. [371]	Specific domain dataset, design like Google search engine	Punctuations marks, language characters, white space, unclassified character gets rid of, special domain ontology based.
Negin Abadani et al. [294]	ParSQuAD dataset, Total 67.73-percent primitively accuracy.	Translating the SQuAD 2.0 dataset makes a current dataset for question answering in Persian applied ParsBERT, ALBERT, Multilingual BERT.
Ali Eshani et al. [308]	70 inquiry for "why" and 20 question for "how", System achieve 68-percent overall accuracy.	Tokenize and POS tagger and text Parse Rhetorical structure theory.
Romina Etezadi et al. [309]	PeCoQ dataset, accomplish total accuracy of 62.75-percent	Knowledge-based approach. Multilingual-BERT and dependency tree for semantic matching.
Nasim Tohidi et al. [290]	Rasekhoon question answering dataset, the outcome was efficient for the actual method analyzed to the existing systems.	Raise the quality of web-based question answering for multi-aim QAs by lexical, contextual, syntactic aspects for Persian language.
Daniel Khashabi et al. [322]	The experiment utilizes multilingual and monolingual pre-trained language models and attempts to handle NLU difficulties in Persian.	Introduced ParsiNLU, comprising manual annotation by native speakers over 14.5k new examples across 6 different Natural Language Understanding (NLU) jobs.
Yasaman Boreshban et al. [194]	Results in an assessment on Rasayel&Massayel corpus show 82.29% accuracy and 56.73% mean related rank.	Develop Rasayel&Massayel Persian question answering dataset. The actual corpus includes 2,118 untrue and 2,051 true inquiries for every single inquiry, text of question.
Ehsan Sherkat et al. [124]	Follow through on quranjooy online automatic QAs, the outcome was standard for the current method.	Present question classification based on rule-based and machine learning ways on closed domain for Persian language.
Naghme Jamali et al. [318]	PerCQA is standard for answer selecting by using multilingual and monolingual pre-trained language models.	Present PerCQA, a first community question answering dataset for Persian Language, which contains 989 questions and 21,915 legal answers.
Ali Mollaei et al. [79]	A classified question on absolute majority voting is utilized to obtain a unique label for single questions.	Propose Persian question classification architecture by applying a Conditional Random Field machine learning model. Test outcomes of many features on its accuracy.
Mohammad Razzaghnoori et al. [210]	The TF-IDF factor should be concentrated cautiously to acquire a good result. The outcome was standard. These inquiries were set up to be hard to answer, even for humans. The efficiency of TF-IDF weighting on raising precision is appreciable.	Word2Vec, which can detain a vast number of exact syntactic and semantic connections utilized to present Persian question classification.
Zeinab Borhanifard et al. [261]	3,600 dialogues about products for online shopping. 65% F1 scores.	Natural language understanding, named entity recognition and ParsBERT for Persian chatbot system.
Jamshid Mozafari et al. [358]	Datasets—PASD, PerCQA, and WikiFA they evaluate PerAnSel result 10.66-percent on PASD, 8.42-percent on PerCQA, and 3.08-percent on WikiFA datasets	deep neural network-based system for Persian question answering

2.2.4 Information Extraction

Mahmoud Rahat et al. [208] describe a method for extracting Persian information based on the Parsa open information extraction system. In this essay, the target language should be able to function independently of its grammar and syntactic structure. The system operates using cutting-edge patterns built on a tree structure. The pattern's goal is to enable Parsa to specify lexical restrictions and POS tags in a way that reduces match. Reduce some challenges posed by the processing of null subject issues and the inaccurate extraction of unused packages for the placement of patterns. The experimentation revealed 2.4 times more promising than state-of-the-art systems in Persian and Baaz, compared to AUC, 1.77 times better than Reverb. The current report calls for more investigation into certain linguistic problems. The author recommends future effort to improve system performance such that tree patterns emerge naturally to upgrade systems. N-array relations are created by the system. Ultimately, the system must improve the Persian dependency parsers' accuracy, especially in complex verbs, and distinguish word relationships in noun phrase.

Raana Saheb-Nassagh et al. [286] describe a method for extracting Persian data based on RePersian, a powerful open data extraction tool for the Persian language. Two different POS tag-sets are used by the system to implement and assess the Persian Dependency Tree dataset Dadegan. The suggested approach is based on certain sentence connection patterns and part-of-speech tagging. The regular expression appearance is used by RePersian to find a semantic link to the right POS patterns that correlate to a connection pattern. The system scored 78.05% medium accuracy for dealing with a single set of arguments, 80.4% for dealing with the second set of arguments, and 56.85% for finding the right dealing among arguments. For future work, the author proposes utilizing the Stanford Token regex of PyRegex for a better searching approach and adding more regexes or modifies to raise the accuracy of precision and recall.

Wei Li [151] provided a method based on shallow parsing of Persian for automated extraction of Persian phrases. The Uppsala Persian corpus was used to use this strategy. The Bijankhan corpus was improved by Mojgan Seraji by deleting, improving, and combining categories. The Bijankhan corpus has been updated into the current corpus. Some errors and inconsistencies in the annotation have been corrected for further updates. The difficulty of parsing can be lowered with the present method. It is crucial for information retrieval based on POS tagging and grammatical rules. The study's findings demonstrate that the present strategy works well on Persian sample sentences for the existing corpus. It is an essential tool for Persian language studies.

Nasrin Rasouli, et al. [209] provided a method for extracting information from ontologies for Persian e-commerce stores. The system displays the products in the form of a table so that buyers can quickly compare the product pricing and quality with another website. The technique is designed to improve recall and precise accuracy. This system's methodology is built on the conceptual model and wrapper, as shown below. For extracting the content of a particular page, wrapper contains a series of highly precise rules. The desired item is extracted from web pages and entered into a database. Conceptual information extraction techniques extract data from free texts based on certain grammatical features. By utilizing a software agent, the precision for user browsing is 90% and the recall is 70%. The technology makes it possible to browse particular items on various websites at the fastest possible pace. It is a smart strategy for more research to work on various languages based on ontologies to speed up the search for users that want information extraction.

Mohammad Mahdi, Hematology et al. [245] provided a method for open information extraction of Persian data. By identifying the dependencies between the words in the phrase and extracting open information, the system builds syntactic labeling associations. Comparatively, most information extraction techniques, the system can extract information with high precision by employing this method.

Mahmoud Rahat et al. [206] describe a method for open information extraction in the Persian language based on the recursive algorithm. The system's operation is based on dependency grammar parsing, which is carried out on the Dadegan corpus using Hazm Python tools. The Persian language benefits greatly from the present instruments. In the analysis, the proposed system showed 68.7% precision. and author compare the system with the English state-of-the-art systems like Reverb[25] 73%, Open Language Learning for Information Extraction (OLLIE) 77%, Stanford OIE 43% precision. Based on our research, it was the best Open Information Extraction (OIE) English system, covering 85% of Stanford Open IE precision-yield curve, making it comparable to systems for English and easily used in downstream applications. Additionally, it outperformed Stanford Open IE in terms of precision. An example is illustrated in 2.5.

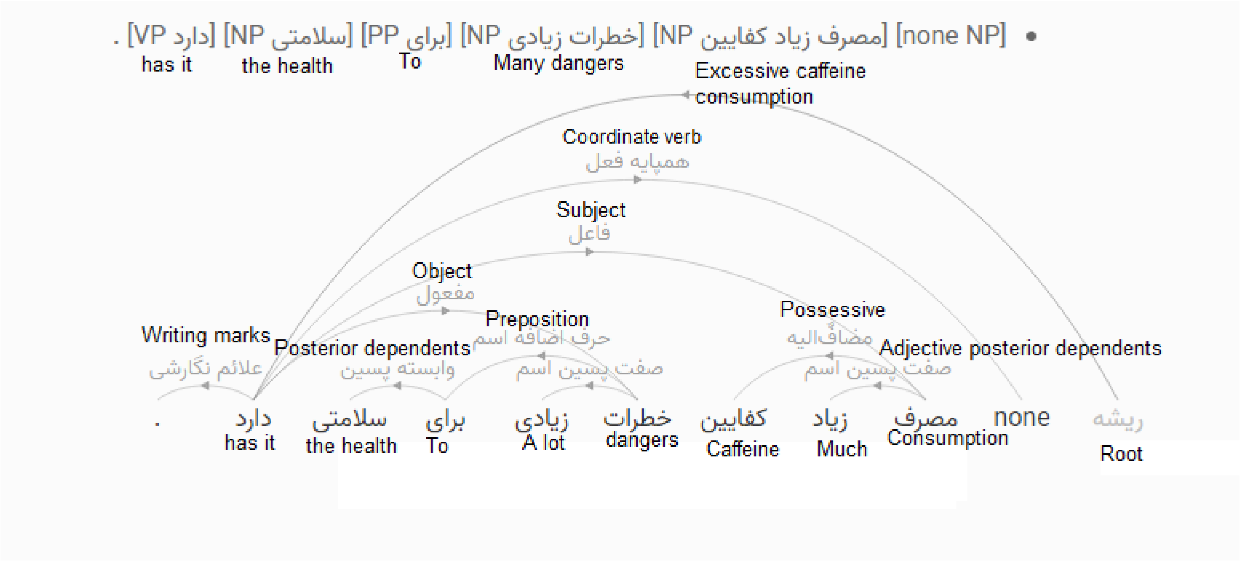


Figure 2.5: Example of a Persian dependency tree generated by Hazm

Mahmoud Rahat et al. [207] provided a method for summarizing Persian literature that uses semantic structure to extract information from the text. The system's operation is based on a text summarizing technique, which is effective for extracting information on demand from sizable databases or online libraries. By tokenizing the sentences into propositions and then finding the most informative ones, the recommended system employed a structurally unique method for text summarizing. Instead of using the complete statement, this method only

[25] http://reverb.cs.washington.edu

uses the necessary portion. When compared to other summarizers, the method produced effective results [354].

Table 2.6: Persian Information Extraction

Paper	Outcome and Datasets	Phases and Approaches
Mahmoud Rahat et al. [208]	Dadegan treebank include of 29,982 sentences, second Uppsala Persian Treebank comprise 6,000 sentences, average precision 59	The ParsPer parser raises the execution of preprocessing steps. Parsa describes POS tags, and lexical analysis limitations to diminish incorrect matches.
Raana Saheb-nassagh et al. [286]	Dadegan dataset, system acquire 78.05-percent medium precision, 80.4-percent for 1 set argument handling, 56.85-percent in second argument considering.	RePersian search for semantic relation in related to right POS pattern to a connection pattern.
Wei Li [151]	Upgraded version of Bijankhan corpus altered by Mojgan Seraji. Uppsala corpus.	Shallow parsing, this algorithm is crucial for information retrieval, machine translation, phrase recognition, the creation of network responding systems, etc. because it can make parsing less complex.
Nasrin Rasouli et al. [209]	Ontology based for ecommerce website, 90% precision on browsing and 70-percent by applying software agent.	Ontology-based Wrapper and conceptual model to raise the accuracy of precision and recall.
Mohammad Madi Nematollahi et al. [245]	Based on current approach, user can obtain information with high precision.	Syntactic labeling correlation by acknowledging dependency relations among words on pulling out sentences. Open information extraction.
Mahmoud Rahat et al. [206]	Dadegan corpus, method acquires 68.7-percent precision, compare to Reverb 73-percent, OLLIE 77-percent, Stanford OIE 43-percent	Preprocessing done by Hazm python tools. Recursive algorithm, dependency parsing[26], [27], [28].
Mahmoud Rahat et al. [207]	Efficient for detecting desired information from large-scale database or digital libraries.	Semantic structure, for Persian text summarization. Tokenizing and identifying the most informatics ones.

2.2.5 Machine Translation

Benyamin Ahmadnia et al. [298] provide a method for improving neural machine translation in the instance of Persian-Spanish by filtering the resulting pseudo-bilingual text. The recommended technique involves the following step: utilizing a model trained for translating monolingual target phrases on a parallel Persian-Spain corpus, translating from Persian to Spain and from Spain to Persian in both directions. Using target sentences from Spain as a reference and synthetic target sentences for both languages as candidates, sentence-level similarity grade is calculated. In this stage, all results for both Persian and Spanish with correlated

[26] https://github.com/yahoo/YaraParser

[27] https://github.com/rasoolims/MSTParserCSharp

[28] http://stp.lingfil.uu.se/ mojgan/parsper-mate.html

synthetic original sentences are arranged in descending order by metric score, and filter sentences with low scores are used. The source-side and target-side of the pseudo bilingual dataset are the filter source synthetic sentences and target sentences for both languages. The target-source model is trained using the best model from bootstrap a to bootstrap b when the filtering stage is complete. Bootstrapping is a technique for enhancing translation performance.

Zohre Komeli et al. [51] look into the issues that Google Translate, Padideh, and Pars encountered when translating from English to Persian. The challenges are demonstrated below.

1. : Lexical Problem: the collection of words is named a lexicon which is Associate to the Syntactic class like a verb, noun, adjective, adverb, etc., and also, transitive/intransitive verbs, generic/proper nouns. EX.: Paper, pencil and pen are the essential ingredients of stationery.

    - Pars: paper, pencil and pen are the essential tools for writing.
    - Padideh: paper draws with pen and pencil are necessary for stationery.
    - Google: paper, pencil and pen are the essential ingredients of chips.

2. Word ambiguity and conjugation: To create new words, morphemes are gathered via conjugation and lexicology. Most often, the meaning of synthesized words is not the same as the word form, and in certain cases, it is unclear in some languages. The words that are synthesized are directly put after one another, without a hyphen or a gap in between. Example: Fruit flies like peach.

    - Google: fruit flies such as peach.
    - Pars: fruit flies alike.
    - Padideh: fruit loves peach flies.

3. syntactic issue: the regulation by which words are set up to make sentences. Example: Don't waste your money on gold.

    - Pars: Waste of your money on gold.
    - Padideh: T-waste your money on gold.
    - Google: Does your money is not on gold.

4. issue in production and transmission level: The examination of the target language's input passage is based on production. The verb-tense point is when the transmission stage difficulty arises. Example, She has been reading the novel before we arrived.

    - Pars: she has been reading the novel before we arrive.
    - Padideh: she has been reading the novel before we arrive, us.
    - Google: she warned she refuse the proposal.

In the period of evaluation of the flowing Translators, contentment about the translators, that can translate 43% of English verbs into Persian and 38% in syntax and conveying meaning. Translators measure with 100 English sentences. An experiment exhibits that translators can translate 50% of sentences to Persian correctly. The good selection in searching for the equivalent 70% in the lexicon.

Benyamin Ahmadnia et al. [166] introduced a pivot language approach for statistical and neural machine translation that enhanced both translation frameworks' quality of translation in comparison to direct translations in the past. In the first scenario, they compared phrase-level combination and sentence-level combination, two popular pivoting translation techniques to the Persian-Spanish Statistical Machine Translation (SMT). By conducting controlled trials, the author evaluated the performance of these two strategies in comparison to a system that was directly taught SMT. The results demonstrated that utilizing English as a bridge language in either approach results in better outcomes than employing a direct translation from Persian to Spanish. Presented a cooperative training plan for the Persian-Spanish NMT, with the second scenario adopting English as a bridge language. Our combined training goal's link word enhances communication between the Persian-English and English-Spanish translation models. The experiments show that this approach significantly improves.

Tayebeh Mosavi Miangah et al. [16] provides a method for translating from English to Persian utilizing the word sense disambiguation. The target language corpus's statistical data was used by the paper's author. Machine translation from English to Persian was used to resolve lexical ambiguity concerns. The results of the statistical method used with the Random Numbers Model for the current paper were quite precise and accurate. The current method needs a straightforward corpus for the target language that has been POS-tagged. A syntactic parser is not required in the source language or the target language. Utilized a monolingual domain-specific corpus in each text of the source domain language to acquire the target language corpus. Any co-occurrence of a word with a number in the target language corpus, the corpus, is statistically important. Accomplished precision for the flowing model is 79%.

Heshaam Feili et al. [13] provides a method for machine translation from Persian to English based on S-TAG, where both languages can use syntactic and semantic elements to change the source of the target language. The author presented an autonomous learning method of the necessary decision trees from a sample dataset for lexical disambiguation selection based on decision tree notation. The inventor of the present approach employed 860 statements totaling 16 words in an experiment. The author creates the XTAG dataset for the English tag. 150 starting trees are selected from a total of, 1227 main trees to be used in the parsing of example phrases. In this paradigm, unlike tenses and individuals, they may translate full sentences with various syntactic structures, such as active/passive forms, by employing a transferred technique.

Abdul Amir Hazbavi [48] presented the results of the English to Persian Translation of Translation Memory (TM). The TOEFL language competence exam was used to choose the study's volunteers, who underwent the author's experiments. By 90 Iranian bachelor's degree candidates in English translation from the Bandar Abbas campuses of Payame Noor University and Islamic Azad University. Subsequently, the topic is divided into two groups of 45 bachelor students each. They tested the results of the translation test scores using the T-test after three experienced translators have translated a text of the same difficulty level. The research's main finding showed that TM tools had a positive influence on translating from English to Persian.

Abbas Ali Ahangar [66] present a model for English Zero-place Predictors into Persian based on the Lexical-Functional Model. The method works grammatical function and C-structure in sentences for noun phrases elaborated in source language (English) in F-structure sentences. Then the same source word was decided for each word. The sentences are generated in the Persian language. The translated sentences shown in table. 2.7.

Table 2.7: Illustration of English Zero-Place predictors into Persian

English zero-place sentences	Translated to Persian
It was night.	شب بود.
It was 6 o'clock.	ساعت ٦ بود.
It was Saturday.	روز شنبه بود.
It was June.	ما ژوئن بود.
It was 2005.	سال ٢٠٠٥ بود.
It is cold.	هوا سرد است.
It was fair.	هوا خوب است.
It is raining.	باران می بارد.
It was snowing	برف می بارید.

Benyamin Ahmadnia et al. [218] a round-trip strategy to support neural machine translation, This round-trip approach's method functions similarly to source-to-target and target-to-source translation. This method was tested by the author when translating from English to Spanish, a language with high resource requirements, and English-Persian, a language with low resource requirements. The present method enhances translation quality following a competitive strategy.

Chris Irwin Davis [70] proposed a Tajiki-Farsi to Persian statistical machine translation. The Tajiki language is identical to Persian. However, it has several dialects and a separate writing system. While Persian utilizes an Arabic-adapted script, Tajiki uses the Cyrillic alphabet. The written form is the sole barrier between these two languages. This issue is solved by the author using part-of-speech tagging and machine translation. Tajiki to Persian translation utilizing mentioned methods. Additionally, Tajiki-Farsi may be used to convert and run Persian language technologies to create new Tajiki lexical resources like WordNet, Frame Nets, and corpora. When they evaluated the performance, they identified two significant underlying causes of errors. Terms borrowed from Russian and Tajik-specific words that are present in the Tajik corpus but do not exist in Farsi. The second is the appearance of Tajik names for things that are not found in Farsi, such as places, people, and organizations with links to Tajik sociocultural entities.

Poorya Zaremoodi et al. [189] incorporated Syntactic Uncertainty using the Forest-to-Sequence Syntactic in NMT. Previous studies found that the syntactic data provided by (naturally error-prone) parsers was promising. To reduce the parser's error, which was caused by the employment of several parse trees like a packed forest, the suggested solution utilized a forest-to-sequence neural machine translation model. It learns the neural attention transmission model from the forest to the target sentences. Studies comparing the sequence-to-sequence neural translation model with the tree-to-sequence models on translations from Chinese, Persian, and English to German provide good results.

Arturo Oncevay et al. [280] provide a method for evaluating the link between languages using two sources—NMT-learned vector and KB—based on Singular Vector Canonical Correlation Analysis (SVCCA), which makes it simple for scholars to comprehend the relationship between languages. The benefits of our approach include the ability to evaluate anticipated languages from only one perspective and the straightforward expansion of language coverage. It is advantageous for multilingual NMT tasks, including language classification and multilingual transfer evaluation.

Chakaveh Saedi et al. [36] provide an automated English to Persian text translation method. The system under consideration is based on PEnTrans, a bidirectional automated translator that contains PEnT1, PEnT1,2 for translation from English to Persian. The PEnT1 is built on knowledge and corpus. Knowledge, corpus, and rules are the foundations of PEnT2. Utilizing Word Sense Disambiguation (WSD) approach, which awards a hybrid collective score with several word senses based on sentences and individual words inside phrases. The accuracy of the present technique for WSD was 91%.

Mehrad Moradshahi et al. [279] provide machine translation-based question-and-answer (QA) semantic parsers. The suggested solution made use of Semantic Parser localizer (SP), which has an eight-language target translation capability for the source language. The benefit of this method is that training data for the target language is generated automatically. For the trial system, the author employed Schema 2QA, a QA system for English on the web, in two domains, first in restaurants, where accuracy rates were 69 percent, and in hotels, where it was 78-percent. The results are excellent when compared to the previous state-of-the-art method, exceeding 40% for restaurants and above 30% for hotels. Spanish, Arabic, German, Persian, Finnish, Italian, Japanese, Polish, Turkish, and Chinese restaurants based on ontologies have been tried.

FattanehJabbari et al. [75] an improved and accessible English-Farsi translation system is developed using the Amirkabir Bilingual Farsi-English Corpus (AFEC). The limited-domain corpus was expanded by the author into an open-domain corpus. Additionally, there are other difficulties like Farsi, unusual written characters, and structural differences between Persian and English. An experiment on the current corpus reveals 9.1% BLEU[29], [30], for the utilized rules and 8.82% BLEU for the normalization approach.

Nicola Bertoldi et al. [91] FBK's machine translation systems can be improved by filing an IWSLT 2013 evaluation complaint. Because of better normalization and tokenization of Persian texts and the acquisition of a sizable monolingual Persian news crawl corpus for English and French as well, the suggested system outperformed baseline English-Persian and Persian-English systems.

Shahram salami et al. [162] provides a phrase-boundary-based method for statistical machine translation. Non-terminal word is labeled by boundary word classes of aligned phrases. In the current study, statistical machine translation was done using a hierarchical paradigm. In addition to POS tags, clustering terms on the target

[29] https://cloud.google.com/translate/automl/docs/evaluate#bleu

[30] https://isl.anthropomatik.kit.edu/pdf/Zhang2004.pdf

corpus are utilized in POS tag to label the rules. It takes several iterations of training and evaluating to identify the ideal number of word classes when using word clustering to define word classes. The suggested model beat the hierarchical phrase-based model with one generic non-terminal due to more accurate tagging. The phrase-boundary model is better suitable for languages with fewer resources than syntactic models, such as those that lack linguistic tools like a parser. For all sentences, identifying non-terminal based on border word classes is simple, and extracting word classes takes less time than extracting parse trees. The suggested model fared better in translation from Persian to French and Spanish to English than the current most advanced hierarchical phrase-based model. This holds true for all languages, including ones with little resources and few linguistic resources.

Mahsa Mohaghegh et al. [41] provided a method for statistical machine translation-based (SMT) English to Persian translation. The initial stage of the present effort is to test how the system converts English sentences to Persian sentences. The current system is implemented on a smart corpus for translation of English to Persian. The issue was fixed by the trained corpus, which was then created for translation. During the experiment, the author realized that if he used his present findings to a large corpus, sentences would be translated with the maximum level of accuracy and in accordance with significant goals or requirements. Utilizing BLEU, NIST, and IBM-BLEU, the experiment with the present system was conducted. The mechanism for translation will need to have its accuracy improved in the future.

Prof. Dr. Zafer Agdelen et al. [249] present a method for English-Persian translating sentences that include homographs, for instance (سبک) sbk and (سبک) sobok both are written in the same character in Persian but with distinct meanings and different grammar like sbk is a noun and sabok is adjective still Google Translate cannot translate such sentence that comprises Homograph in the Persian language. To find homographs in Persian sentences and convert them to target words like English and other languages, the author of this study proposed grammar based on Persian terms. The verb is found in the Persian corpus using a stemming technique for the Persian language. The trial showed that the existing system was overly effective in comparison to alternative approaches. This approach has realistic accuracy while identifying not just nouns or verbs, but also adverbs and adjectives.

Mohammad Iman Askari et al. [300] provide evaluation tools for texts that have been translated from English to Persian based on Google machine translation and human translation utilizing the Translation Quality Assessment (TQA) approach in an academic setting. The method assesses and contrasts the two translated versions, and the resultant output reveals that the English to Persian translation has several subliminal and overt flaws. Universities and academics can evaluate the caliber of translation using the existing approach, which has broad applicability. For evaluating the other language pairings, the author of the current work advises utilizing the TQA application or other corresponding programs.

Marziyeh Taleghani et al. [251] provide assessment or evaluation tools based on Translation Evaluation Metrics for English to Persian translation phrases (MTEMs). Several languages use the existing system. This study tries to rate the accuracy of the Persian text's machine translation of its lexical likeness. The author employed a dataset of 200 English sentences in the experiment phase that were translated using a different method from statistical machine translation and matched to human translations. Three human evaluators and the seven MTEMs independently evaluated the results. Humans and MTEMs used Spearman and Kendall coefficient correlations to calculate the results. In the end, the analysis revealed that GTM was more reliable than NIST and BLEU measurements.

M.shamsi et al. [337] offered a method for converting sentences from the Persian language into sign language (PSL). Use the lexical transform module to aid blind and deaf people who are disabled by converting Persian phrases into signs. Normal people can communicate with them to solve their issues because of the existing technology, which enables them to interpret human signs. The system employed an avatar to display translated phrases in which words were converted to HamNoSys notations, rotated HamNoSys script into SiGML, and then converted SiGML to 3D character animation using the language. Altering the animated character's avatar's facial expressions is another option.

Amin Mansouri et al. [77] gives a comparison of several statistical machine translation algorithms used for Persian to English translation. The primary accomplishments of the current research are the suggested semi-automated way to extract a parallel corpus for the English-Persian language from translation books. The MaxEnt classifier is utilized as the foundation for a noise filtering technique to identify the sentence mistake pairings that can be identified by aligning the sentence stages. 400 sentences and an EGIU exam that was found in an English Learning book are included in the current system evaluated with PCTS. In terms of the BLEU metric, verb-aware SMT results in state-of-the-art English-Persian translation.

Abulfat Fatullayev et al. [365] provide a dictionary of Persian-Azerbaijani translations based on corpora and statistical research. To overcome the existing corpora collected from newspaper scientific sites, social, political, economic, sports, etc., on homograph terms, an alternative corpus was built. This dictionary's initial edition has 21,000 words and 3,000 phrases in Persian and Azerbaijani.

Nava Ehsan et al. [92] a statistical machine translation system was used to offer grammar and syntactic error checking for Persian and English. The suggested method is useful for electronic text, spelling and grammar checks, and sentence transformation from unfiltered to filtered sentences. In certain English and Persian texts, the experiment reveals gains of roughly 24% in the ability to correct and identify syntactic and grammatical problems.

Behzad Mirzababaei et al. [99] provide a spell-checker in discourse-aware statistical machine translation employing context-sensitive translation. The suggested solution improved the SMT-based real-world error detection. Choosing the best candidate using discourse analysis, such as from an animal or car. It helps to maintain the relationship between the words used in a candidate's phrases and the keywords used throughout the whole document. The PMI measure of word distance was used to identify dependencies in documents. SVM-rank was used to determine the weights for each feature by logarithmically reranking the N-best results.

Mitra Shahahbi et al. [37] provide a rating on the accuracy of the Persian machine translation work. The largest English-Persian MT software: Padideh and Pars translators are used in the assessed model. The top result is generated by the author with the Padideh MTs. These two machine translation tools implement complex phrases. Morphology, syntactic ambiguity, Persian semantic analysis, and protracted phrases are taken care of. In terms of syntactic and semantic analysis, the given issue is complicated. Lexical coverage, phrases, short sentences, and to a lesser extent morphology were areas where both MTs algorithms excelled. However, the performance was not the same in terms of grammatical ambiguity, long phrases, and target language generation. Mohammad Taher Pilevar et al. [43] provide a method based on statistical machine translation for translating English to Persian, the first effort. The suggested system operated using phrase-based models such as p(e/f) probability 'e' for the target phrases and probability 'f' for the source language sentences. In the experiment, the author employed a small parallel corpus in addition to a small corpus built from movie subtitles. The outcome was pleasing. To produce a more accurate translation of English to Persian, a vast corpus of bilinguals is needed.

Ahmed El Kholy et al. [93] offered a phrase pivot SMT-based method for translating from Persian to Arabic. The lack of parallel corpus is the fundamental barrier to statistical machine translation. This study suggests employing a third language to develop a new statistical machine translation system for language pairings. For instance, machine translation (MT) is accessible from Persian to English and English to Arabic, but not from Persian to Arabic. The author offered the Pivot phrase strategy for translating phrases from English to Arabic and Persian to English as a solution to this issue. they can build a parallel translation corpus from Persian to Arabic using this method. The system indicates that the translation has improved by 0.6 BLEU points of Persian to Arabic SMT.

Mahsa Mohaghegh et al. [78] developed a method based on hierarchical statistical machine translations for translating English to Persian. The suggested approach translates from Persian to English and from English to Persian in both directions. The author employed numerous parallel corpora with Moses during the experiment phase. Joshua is a hierarchical tool set, but Moses is not. Moses' performance had a higher BLEU and NIST score than Joshua's. Joshua has a greater sense of word order than Moses. According to the study, good outcome has been observed in employed language pairings. Mohaddeseh Bastan et al. [169] developed a method for translating from Persian to English using neural machine translation with few resources. To improve word alignment, the author of this research applies a preprocessing step and a novel loss function. The system obtained a 1.87 BLEU score, which represents a 0.9 improvement for transliteration jobs and a good increase in translation quality. Yasaman Motazedi et al. [35] an automated method for translating from English to Persian is called PEnT1 or PEnTrans. The suggested system translates phrases in English with 12 tenses and an active, passive, interrogative, and negative verb using rules-based and semantic-based methods. With the present strategy, the author uses the word sense disambiguation hybrid technique to compute the score, and the assessment reveals that the system performs well when compared to other systems.

Mahsa Mohaghegh et al. [52] enhance the accuracy of domain adaptation studies using statistical machine translation. The suggested approach took advantage of a noteworthy dataset to translate sentences from English to Persian and to improve sentence alignment and matching. For improved outcomes, the system makes use of Moses' decoder technique. The author also emphasized that a large corpus is not necessary to improve performance. Instead, putting greater emphasis on the corpus's domain helps to increase SMT's accuracy. Comparing the existing system's performance to that of Google Translate and the German-Chinese SMT languages, it is a decent option.

Mahsa Mohaghegh et al. [33] worked on expanding the statistical machine translation training corpus for English-Persian. In the initial test, the SMT method was used to translate from English to Persian to pinpoint and address issues with the output that the system produced. The author tries to increase the size of the parallel corpus for efficient translation in the second section. The author's conclusion was that the BLEU score would rise along with the corpus size. The primary contribution of this research is the necessity for a corpus size that is proportionate to the effects of translation. The author advises rearranging the sentences for even greater benefit.

Alireza Mahmoudi et al. [97] developed a method based on Rule-Based Machine Translation (RBMT) and statistical machine translation for translating a richly morphological language like Persian to a poorly morphological language like English (SMT). The first experiment classified the mistakes using rule-based MT, then applied SMT for rich morphology using bilingual corpora. The current method demonstrates that verb conjugation problems in Persian language morphology present numerous difficulties in improving the accuracy of RBMT. They begin by creating inflected verbs from the most common feature values. An enhanced rule-based system with 16K test set phrases revealed a 2.6-percent BLEU score.

Marlies van der Wees et al. [215] test machine translation efficiency by using 4 genres which are collected from public websites for four different languages. The proposed system implements Chinese-English, Arabic-English, Bulgarian-English, and Persian-English with conversation, leading articles, news, and speech genres. With genre classifiers by Support Vector Machine (SVM), Linear kernels. The result shows an Arabic-English 17.9 BLEU score, Chinese-English 13.9 BLEU score, Bulgarian-English 33.4 BLEU score, and Persian-English 22.3 BLEU score, which shows the best performance compared to single genre-agnostic systems. Finlay shows 100% accuracy with 4 genres. The system can also adopt Neural MT for future work.

Homa Fadaei et al. [310] based on New Mark's categorization and translation methodologies from 1988, it was suggested to investigate both human and machine translation in the Khalid Hosseini book The Kite Runner. The proposed study is a review of the novel The Kite Runner, which chronicles the life of an Afghan child named Amir, who lived in Wazir Akbar Khan in Kabul. To locate the culturally distinctive information in writings that have been translated from Persian to English. The experiment demonstrates that while machine translation was altered, the human translation version preserved the novel's uniqueness. Two Persian translators, Ganji-Solymanzadeh (2004) and Ghabraye (2003), were ultimately responsible for the translation's completion.

There are 100 pages of frequency percentage results in the corpus of Kite Runner. Human translation had 100% accuracy, but in machine translation had 75%.

M.Zahedi et al. [165] developed a method based on statistical machine translation for identifying multipart words. The suggested approach is built on a parallel corpus, where the source language for one side of the corpus is multipart words that have not been modified. An altered multipart word in the target language can be found on another side of the corpus. The phrase-based hierarchical model was added with parameters from a Fertility-based IBM model[31], [32], utilizing Synchronous Context-Free Grammar (SFG) to extract linguistic information. The syntax-based decoder employed in the experiment for the submitted system helped to identify space between Persian words and corrected according to training data. The outcome ultimately met the demand.

Ruhi Sarikaya et al. [28] suggested a method for continuous space Persian to English translation. The Tied-Mixture Machine Translation (TMMT) underlies the submitted system's operation. The system has continuous parameter translation capabilities. As tools for training the off-the-shelf speech recognition acoustic model, parametric tables were employed. The outcome of the existing strategy was driven. According to an experiment, phrase-based TMMT has a high BLEU score, but word-based TMMT had a lower score when compared to phrase-based TMMT.

Mina Zand Rahimi et al. [188] presented a study comparing current Google Translation and Google Neural Machine Translation to English to Persian translation. Compare the two translations to Keshavarz human examination model. The experiment's findings demonstrated that Google's neural machine translation is effective in terms of successful semantic translation from Persian to English.

Benyamin Ahmadnia et al. [219] author provided a method for concurrent low-resource language statistical machine translation using the round-trip training approach. With the suggested method, low-resource languages like Persian-Spanish and high-resource languages like Spanish-English may both be translated. The experiment demonstrates that the round-trip training strategy can handle the scarcity of training data while still enhancing translation performance. The performance outcome of round trip is higher than self-training and baseline approach; round-trip exhibited good performance with high resource language like English-Spanish compared to a low resource language like Persian-Spanish illustrated in table 2.8.

Table 2.8: The round-tripping system result comparison

Translations Systems	Spanish-English	English-Spanish	Persian-Spanish	Spanish-Persian
baseline	23.51	24.33	18.64	17.42
self-training	22.96	24.42	19.22	17.89
round-tripping	27.83	28.11	22.14	21.02

Benyamin Ahmadnia et al. [143] offered a method for contrasting Persian to Spanish translations done directly

[31] https://www.jstor.org/stable/2137661

[32] https://aclanthology.org/P97-1022.pdf

with Persian to Spanish translations done using English as the pivot language. In comparison to straight translation, the outcome demonstrates the great performance of a pivot technique, with a BLEU score of 1.12.

Ahmad El Kholy et al. [94] provided a method for Persian to Arabic translation using a careful combination of Direct and Pivot Statistical Machine Translation. The suggested approach improves the BLEU score on the distinct direct training corpus from 0.4 to 3.1. The pivot translation model exhibits a significant drop in selected mixing. But the primary problem is that this does not boost performance. The author advises investigating towards pivot model identification based on morphological patterns obtained from the direct model rather than only the specific surface shape in the future to get around this issue.

Mohammad Sadegh Rasooli et al. [101] provided a method for statistical machine translation from Persian to English based on morphological and orthographic processing. The suggested technique demonstrates how splitting apart a Persian verb may enhance the accuracy of Persian-English translation, with a 1.9 BLEU Point improvement in a blind test set. Although it is left for future work, the author experienced word-ordering problems during the experiment. The author recommends creating a Persian dependency tree bank to enhance word order.

Nasrian Mustafian [179] offered a human and automatic evaluation of Persian to Swedish Machine Translation mistakes, as well as automatic post-editing for Persian Machine Translation. Persian Machine translation error appears on three datasets using the suggested strategy, according to the author's experiment. A recommended system employs the manual application of the Automated Post-Editing (APE) principle. Mistranslated words, prepositions, and incorrect word order were the errors. The inaccuracy in question was categorized as syntactic and semantic, respectively. There was little connection between automated and manual evaluations. Implementing APE in BLEU rate on a big corpus remains the same, however, the TER score improves on a single sample. However, the other dataset's score remains unchanged.

Saeed Farzi et al. [131] provided a method for statistical machine translation of word order based on a phrasal dependency tree. The suggested model was employed in the context of a log-linear model, which used statistical and syntactical data to solve reordering issues. The performance of the existing system has improved by BLEU/TER 1.54/1.7, 1.98/3.01. Persian-English and English-German translations are done using the bilingual Tehran corpus. In the 2007-Second Statistical Machine Translation Workshop Benchmarks (WMT07).

Somayeh Bakhshaei et al. [39] provide a method for translating German to Farsi and Farsi to German using bridging language as the basis for statistical machine translation. Because there are numerous resources available for the Farsi-English and German-English language pairings, the suggested system uses English as a pivot language for translating from Farsi to German. The method concludes by displaying a BLEU score of 15% improvement in the Farsi-German translation.

Masood Hamed Saghayan et al. [333] gave a way of figuring out just how much machine translation affects the ability to spot fake news in Persian COVID-19 tweets. The suggested approach uses a Persian dataset of roughly 2 million tweets that was gathered from Twitter. The Support Vector Machine (SVM) classifier was applied to extracted tweets. The author then used a machine translation technique to translate the Persian dataset into English, a language with numerous resources. The results demonstrate that machine translation has a detrimental influence on binary accuracy, classification is 4% negative, and multicast classification is 23%

negative, respectively.

Mohammad Mahdi Mahsuli et al. [176] offered a method based on an attention-based technique for translating from English to Persian using deep learning. Every word in the suggested system was converted into a numeric vector using word2Vec. Deep neural networks are employed in this approach to identify several meanings of words in a training corpus by vectorizing characters. In the experiment of the present system, which uses a corpus of, 14,000-word pairs, performance increase using the attention-based technique is higher than the 4.21 BLEU score.

Hadis Ghasemi et al. [148] offered a comparison of Persian to English and English to Persian to identify errors in Google Translation. The suggested approach utilized error analysis based on Keshavarz (1999), which examines errors pertaining to this significant group. A phonological mistake, a Lexico-semantic mistake, a syntactic-morphological mistake. While other frequencies differ, the error frequency for active and passive voices is equal to 2. In conclusion, the chi-square comparison demonstrates that direct translation is not advantageous for high-quality Google translation.

Mahdiyeh Alimohammadi et al. [127] provides a method for applying statistical machine translation in online communication between hearing and deaf youngsters. Moses machine translation was employed. For youngsters to converse online, there is enough information in the Persian sign language corpus, according to the domain. Aim to produce a video dataset in the future that has 200 webcam recordings of three deaf people. The present technique will also be utilized for complete online connectivity between youngsters who are deaf and those who are not.

Hosein Azarbonyad et al. [88] offered numerous translations for information retrieval in English and Persian (CLIR). Query ambiguity is the key problem affecting CLIR's performance. To translate a query, the author used two context-based approaches to extract phrases from multilingual corpora. An experiment demonstrates the effectiveness of the phrase-based strategy for English-Persian cross-language information retrieval. Results improved by 106% when using a corpus and by 7% when using a bilingual corpus. The author suggested using contextual data for future studies, such as the mutual information of candidates' translation in query process translation.

Fahime Mohammadpour et al. [370] provided a method for an English to Persian hybrid machine translation system. The suggested system employed two methodologies: rule-based and corpus-based. The outcome of the experiment demonstrates how many natural phrases may be transformed using the proposed strategy and proper Persian word order.

Abdol Hamid Pilevar et al. [56] provided a method for improving rule-based machine translation quality using statistical post-editing. To automatically translate the output of rule-based machine translation, the author employed SPE and the Moses toolbox. For comparing the performance results of various systems, BLEU, and TER metrics were employed. The outcome demonstrated that the post-editing module would apply to RBMT as a new domain.

Mina Abbaszade et al. [295] proposed a quantum natural language processing-based machine translation of English to Persian. The suggested system makes extensive use of quantum long short memory (Q-LSTM), particularly for translating English phrases into Persian. Other languages can also be translated using the recommended technique. The experiment demonstrates that the present methodology is more accurate and quicker than traditional methods.

Heshaam Faili [22] showed the development of a word meaning disambiguation using machine translation from English to Persian. In the suggested approach, Word Sense Disambiguation (WSD) was detected using decision trees that were learned autonomously from training datasets. The NIST measure saw a 16-percent improvement as a consequence. In the WSD experiment, 81% of the ambiguous words in the corpus of 860 aligned sentences were right.

Jan w. Amtrup et al. [10] finite state transducers and feature description structure were used to improve the effectiveness of morphology in machine translation. For accurate morphological analysis and generation, the method is particularly successful. The finite-state morphological system is simple to construct and has high runtime performance. In a large corpus, the system is also effective in locating and combining the appropriate morpheme for the word.

Mehrdad Vasheghani Farahani [377] compares and contrasts the effectiveness of human and machine translation. The experiment uses seven corpora that represent various genres. Machine translation from Google and Bing was employed. The material is being translated back into English by two qualified human assessors. When it comes to punctuation marks, human translation outperforms Bing and Google by 64.6%. Accordingly, the machine translation yielded 54.4% and 53.6%.

Ebrahim Ansari et al. [106] provided a method for constructing a bilingual corpus from Wikipedia for Persian to English utilizing bidirectional translation. To improve statistical machine translation, the author of the suggested method employed Wikipedia as a distinguishing corpus to extract bilingual phrases in Persian and English. Other low-resource languages can likewise use the existing system. Bidirectional novel translation-based information detection is used to calculate the similarity score between two phrases, and a length penalty score is utilized to improve the accuracy of the extracted corpus. The outcome demonstrates that the current strategy may significantly enhance Persian-English translation quality.

Sanaz Kokabpeik et al. [369] offer a survey on the satisfaction of English as a Foreign Language (EFL)[33] students on the reliability of Google Translation for use in critical academic-related work in English and Persian. 100 people, 100 of them male and female, completed a questionnaire with 12 questions that the author had written. The majority of answers for EFL users have been satisfied that the Google translation system is not accurate enough to be utilized in academic areas, and all answers have been incorporated into Statistical Package for the Social Sciences (SPSS) version 23[34] software with chi-square assessment and bar charts.

Somayeh Bakhshaei et al. [67] develop an approach for improving the Farsi-English Machine Translation System based on pool-based active learning. The authors utilized an active learning method for random sample selection and used features for the best case to have a good performance. Utilizing the current method, they can raise the baseline system at a low cost, with just inquiring for the translation of influential sentences for system quality improvement.

Mohammad Reza Mirsarraf et al. [98] presented an approach for evaluating a machine translation system based on dependency-inspired semantics. The proposed system is designed to assist the output of English-Persian

[33] https://www.teachingenglish.org.uk/article/efl

[34] https://www.ibm.com/support/pages/downloading-ibm-spss-statistics-23

machine translation using dependency grammar for checking sentences semantically. The goal of the evaluation was to not change the actual meaning of the sentence after translation. In experiment, result shows that in the current approach, evaluation rate correlated better and significantly with human judgment.

Milad Maleki et al. [177] developed a method for comparing human and machine translation in the translation of Ellipsis from English to Persian. The suggested approach was used in the MIZAN corpus, which contains one million phrases translated for text matching with Google Translate. The outcome reveals some verbal and nominal ellipsis differences between English and Persian, but no clausal ellipsis differences were discovered. The quality of the human and machine translations differ noticeably. The research also demonstrates that human translation still has a qualitative advantage over machine translation.

Benyamin Ahmadnia et al. [254] provided a method for improving concurrent English-Persian neural machine translation in low-resource environments. After researching several neural network optimization strategies, the present approach was developed. The outcome confirms that using the present approach for English-Persian translation is efficient and much superior to the English-Persian SMT model in terms of quality.

Benyamin Ahmadnia et al. [297] improves the accuracy of Neural Machine Translation between Spanish and Persian. The current approach used attention-based neural machine translation (NMT) to predict and examine translation performance. When fluency and sufficiency are considered, the findings are also reviewed using a human translator in addition to the BLEU score. The outcome demonstrates that employing a large training corpus improves NMT performance. Named entities are frequently mistranslated by NMT. The caliber of the translation demonstrates the system's dependability as a substitute for human translation.

Hakiminejad Afshin et al. [142] compared verb tense accuracy between Google Translate and human translation for English to Persian machine translation. Research indicated that Google Translate cannot accurately translate verb tenses from English to Persian. Additionally, numerous grammatical mistakes are created by Google Translate during the evaluation of translations. These mistakes depend on aspects, passive, compounds, and lexical analysis.

Reza Khanmohammadi et al. [321] developed a method based on a revolutionary neural machine translation methodology for converting prose into Persian poetry. The present approach uses transformer-based language to translate the great Persian poetry. To get the final translation, the author tried to create a transformer model from scratch and pre-train several modifications of BERT. The current approach is successful at producing both professional and unprofessional Persian poems, according to the results.

Shahram Salami et al. [161] presented a brand-new hierarchical model for statistical machine translation has been presented that labels non-terminal by boundary word classes of aligned phrases. The proposed system used probabilistic synchronous context-free grammar. Part of Speech tags and word clusters trained on the target language corpus are used for the coarse-grained and fine-grained non-terminal used for labeling the rules. The author has also put forth a fresh method for filtering rule extraction that is based on phrase pair alignment. The extraction of hierarchical rules is constrained to phrase pairings that may be divided into two aligned sub-phrases using limited patterns of rules. The model size and decoding time are significantly reduced by the suggested filtered rule extraction, with little to no effect on the translation quality. The suggested approach significantly outperformed the leading hierarchical phrase-based model in our studies using BLEU as a metric for the translation of Persian, French, and Spanish into English. This holds true for all languages, including those with little resources and no linguistic tools.

Nicola Bertoldi et al. [90] enhanced French-English, Persian-English, and English-Persian in FBK's machine translation while assessing the 2013 IWSLT campaign. The author enhanced the text normalization and constructed a sizable monolingual news corpus for Persian to improve the Persian-English and English-Persian approach above the baseline provided by the workshop organizers.

Razieh Esmailpour et al. [268] proposed a method for translating a user's question into Persian. Using a natural language processing strategy, the suggested technique Query Expansion Assistant database (QEAD) is utilized to translate title references through worded extraction. Google Translate and Google Scholar were used by people to evaluate the system. The initial evaluation performance reveals 60% performance similarity and 40% performance dissimilarity between GT and QEAD. In the end, the system performed better in the test.

Behrang QasemeiZadeh et al. [17] MULTEXT-East Framework is used to develop writing systems and morphological analysis tools. The system has a special part-of-speech tagging categorization and is developing a single Farsi orthography for digital environments, as well as an annotated corpus and vocabulary that are under preparation.

Jan W. Amtrup et al. [7] review the Shiraz project, a computerized Persian to English translation. A machine translation system uses unified and typed feature structures. An introduction of Persian linguistic characteristics is provided, as well as the morphological and syntactic grammar developed as part of the Shiraz project. Using a chart, the approach offers an integrated view of the results produced at all levels of linguistic description. The perspective of the descriptions themselves is unified by the usage of a complex typed feature structure formalism [354].

Table 2.9: Persian Machine Translation

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Benyamin Ahmadnia et al. [298]	Pseudo bilingual dataset. Bootstrapping method for raising the performance of translating, Persian-Spanish on both direction.	Raising neural machine translation by filtering generated pseudo bilingual text in case of Persian-Spanish.
Zohre Komeli et al. [51]	Translator can only translate, 43-Percent English verb into Persian, 38-percent in syntax and conveying meaning, 50-percent accurate sentences to Persian on evaluation of 100 English sentences, seeking same in lexicon is 70-percent	To Look into the difficulties of Google Translate, Padideh and Pars machine translation during translating English-Persian.
Benyamin Ahmadnia et al. [166]	English as a bridge language can raise the quality of translation from Persian-Spanish.	Pivot language technique

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Tayebeh Mosavi Miangah et al. [16]	Monolingual domain specific corpus, reach precision of 79% for actual model in English to Persian translation.	Word sense disambiguation, POS tagging and syntactic parser, Random number statistical approach.
Heshaam Feili et al. [13]	XTAG dataset choosing from 150 first trees from all 1227 primary tree. System can translate syntactic structure like active/passive forms unlike Person and tenses	S-TAG established for translation of Persian to English, semantic and syntactic
Abdul Amir Hazbavi et al. [48]	Demonstrate translation memory is helpful for English-Persian translation. Assessment on human and computer translation of text.	Based on 3 skillfulness translation raters, T-test
Abbas Ali Ahangar et al. [66]	Translate the Zero-place English sentence into Persian.	Utilised Lexical-Functional grammar model.
Benyamin Ahmadnia et al. [218]	Source-target, and target-source translations. Assessment done on English to Spanish as advanced resource and English-Persian as lesser resource language. The system can enhance the caliber of translation.	Round-trip approach with a reinforce neural machine translation.
Chirs Irwin Davis [70]	TAP corpus, Bijankhan corpus, system obtains 94.98% in recognized words, 63.22-percent in unrecognized words, and total accuracy of 92.52-percent.	Used phrase based statistical machine translation, with execute part-of-speech tagging and machine translation, two different tasks for Tajiki-Persian translation.
Poorya Zaremoodi et al. [189]	Assessment done on Chinese-Persian, and English-German translation shows Perfect Result over Sequence to Sequence neural translation model and tree to sequence model.	Forest-to-Sequence model, an unpredictability in neural machine translation for low resource to high resource language.
Arturo Oncevay et al. [280]	Helps researchers to realize relationship between languages.	Singular Vector Canonical Correlation Analysis (SVCCA).
Chakaveh Saedi et al. [36]	PEnT1, PEnT1,2 corpora, system obtain 91% accuracy for WSD	PEnTrans is a bidirectional automatic translator.
Mehrad Moradshahi et al. [279]	Schema2QA, QA datasets for English on the web in two areas, systems acquire 69-percent accuracy for restaurants and 78-percent for hotels. In localised language like Spanish, Arabic, German, Persian, Finnish, Italian, Japanese, Polish, Turkish. Chinese tested.	QA semantic parsing dataset using machine translation. Semantic Parser Localizer (SP) to translate source into 8 different languages.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Fattaneh Jabbari et al. [75]	Amirkabir Bilingual Farsi-English corpus (AFEC), the system acquire 8.82-percent BLEU for normalization method 9.1-percent BLEU score.	Increased an open English to Farsi translation system utilizing Amirkabir Bilingual Farsi-English Corpus.
Nicola Bertoldi et al. [91]	For bettering normalization and tokenization of Persian text, applied large monolingual Persian corpus gathered from Persian News. For translation, English-Persian and Persian-English system surpass a workshop provided baseline.	Enhance FBKs Machine Translation System through IWSLT 2013
Shahram Salami et al. [162]	In translation from Persian-French, and Spanish-English system outperform comparison to state-of-the-art hierarchical phrase-based model.	Statistical machine translation, Phrase-boundary. POS tags by clustering word on the target corpus.
Mahsa Mohaghegh et al. [41]	Smart corpus for translation of English-Persian, An assessment that reveals a big corpus can improve accuracy.	Statistical Machine Translation.
Prof. Dr.Zafer Agdelen et al. [249]	The system not only find noun or verb but also discover adverbs and adjectives with credible accuracy.	English-Persian translating sentence that have homograph, Grammar based.
Mohammad Iman Askari et al. [300]	TQA application is good for evaluating the quality of translation.	Measure English-Persian translation text genre using Translation Quality Assessment (TQA) model within academic context.
Marziyeh Taleghani et al. [251]	A dataset of 200 English sentences that correlate to human translation. The outcome acquires by human and MTEMs are calculated by Spearman and Kendall coefficient correlation, GTM demos trusty compare to NIST and BLEU measure.	Translation Evaluation Metrics (MTEMs) for judgment of English-Persian.
M.Shamsi et al. [337]	HamNoSys Notation.	Convert a Persian phrase to sign language (PSL). The lexical transform module is used to translate Persian phrases into sign language.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Amin Mansouri et al. [77]	PCTS consists of 400 sentences and an EGIU test. In terms of the BLEU metric, verb-aware SMT produces cutting-edge results for English Persian translation.	Statistical machine translation, semi-automated method. MaxEnt classifier for noisy filtering.
Abulfat Fatullayev et al. [365]	There are 21,000 words and 3,000 phrases in the Persian-Azerbaijani lexicon. Make a parallel corpus to overcome homograph terms.	Statistical analysis approach for Persian-Azerbaijani translation dictionary.
Nava Ehsan et al. [92]	The results reveal a 24% improvement in repairing and eliminating syntactic and grammatical errors in certain English and Persian material.	Statistical machine translation framework
Behzad Mirzababaei et al. [99]	The system is useful for retaining the link between a word in a candidate sentence and the entire document, spell checker system.	Discourse-aware statistical machine translation, Utilizing PMI measurement for having weights of each feature, SVM-rank by log-linear re-ranking the N-best Result.
Mitra Shahahbi et al. [37]	The assessment approach used on Padideh and pars two Translator on English-Persian. Both systems performed well in lexical coverage, phrases, and short sentences, but not in syntactic ambiguity, lengthy sentences, or target language production, Padideh was the best.	Assessment of quality among Padideh and Pars translator.
Mohammad Taher Pilevar et al. [43]	The result of a small parallel corpus derived from movie subtitles was satisfactory for English to Persian translation.	Phrase-based model, statistical machine translation.
Ahmad El Kholy et al. [93]	Create a parallel translation corpus from Persian to Arabic using English as the pivot; the system returns a BLEU score of 0.6 in Persian to Arabic SMT.	Phrase Pivot SMT.
Mahsa Mohaghegh et al. [78]	Many parallel corpora using Moses, Joshua, which is a hierarchical toolkit, BLEU, and NIST score show that Moses outperformed Joshua.	Hierarchical Statistical Machine Translation.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mohaddeseh Bastan et al. [169]	The system earns a 1.87 BLEU score on the Verbmobil English to Persian translation dataset and a 0.9 improvement for the transliteration challenge.	Scarce-resource neural machine translation, implement preprocessing and new loss function for word alignment enhancement.
Yasaman Motazedi et al. [35]	System translation sentences in English with 12 tenses, passive, active verb, interrogative phrases, and negative.	Machine translation using statistics: To generate the score, the phase-based model additionally employed a mix of word sense disambiguation.
Mahsa Mohaghegh et al. [52]	When compared to Google Translate and German-Chinese SMT language, the existing system performs well.	Enhanced Statistical Machine Translation quality in domain adoption experiment. English to Persian. Moses decoder provided better result.
Mahsa Mohaghegh et al. [33]	The primary contribution of this research is improved sentence alignment.	English to Persian training corpus expanded. Machine translation using statistics.
Alireza Mahmoudi et al. [97]	bilingual corpus, enhanced rule-based shows 2.6% BLEU score with 15K test set sentences.	Translating rich morphological to poor morphological Persian-English by using of RBMT and SMT approach.
Marlies van der Wees et al. [215]	Trial 4 genres, which are gathered from open websites for 4 distinct languages. Outcome reveals Arabic-English 17.9 BLEU score, Chinese-English 13.9 BLEU score, Bulgarian-English 33.4 BLEU score Persian English 22.3 score, eventually demos 100-percent accuracy with 4 genres.	Support vector machine, linear kernels.
oma Fadaei et al. [310]	Kite Runner Corpus, human translation accuracy is 100-percent and machine translation 75-percent.	Newark (1988) classification. For investigating human and machine accuracy.
M.Zahedi et al. [165]	Parallel corpus, the outcome was according to necessity for finding a multipart word.	Statistical machine translation, fertility-based IBM model for probability distribution. Used Semantic Frame Generator (SFG) in phrase-based hierarchical model, and syntax-based decoder for finding space between Persian words.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Ruhi Sarikaya et al. [28]	Outcome presents BLEU score is high for phrase-based TMMT but for word-based was low in contrast to phrase-based TMMT.	Tied-Mixture Machine Translation (TMMT) for translating continues parameters, off-the-shelf speech recognition acoustic model training tools.
Mina Zand Rahimi et al. [188]	Human investigation model of Keshavarz. The results suggest that Google neural machine translation is useful for translating from English to Persian.	Assessment of old Google translation and Google neural translation.
Benyamin Ahmadnia et al. [219]	The results suggest that the round-trio training strategy is helpful for dealing with the scarcity of training data and boosting translation performance.	Statistical machine translation with Round-trip Training for translating high-resource Spanish-English and low-resource Persian-Spanish languages.
Benyamin Ahmadnia et al. [143]	Outcome demos BLEU point of 1.12 in contrast to direct translation.	Direct translation of Persian to Spanish vs. Translation of pivot languages using English as the pivot language.
Ahmad El Kholy et al. [94]	In-house parallel corpus of around 165k sentences and 4 million words, the selected compound of direct and pivot translation model guides a large deduction of the pivot model and also lacks performance efficiency. On several direct training corpora, the system achieves a good BLEU score, ranging from 0.4 to 3.1.	Pivot-based composite method and direct statistical machine translation (SMT).
Mohammad Sadegh Rasooli et al. [101]	Improve the quality of Persian-English translation with a 1.9 BLEU score.	Orthographic and morphological processing was used.
Nasrian Mustafian[179]	Implemented on three datasets, the results demonstrate a poor correlation between automatic and manual evaluation. The TER score improves on one dataset, while the scores on the other datasets remain the same.	Assessment of Persian-Swedish MT mistake manually and automatically, Automatic Post-Editing (APE).
Saeed Farzi et al. [131]	Tehran bilingual corpus using WMT07 benchmarks. Result show BLEU, TER 1.54, 1.7, 1.98, 3.01 advances in Persian-English, English-German translation.	Statistical machine translation, phrasal dependency tree, and log linear model are all methods for overcoming word reordering difficulties that use a combination of syntactical and statistical information.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Somayeh Bakhshaei et al. [39]	The results reveal a 15% improvement in the BLEU score for Farsi-German translation.	Based on a bridge language that uses English as a pivot language for both sides of translation from Farsi to German.
Masood Hamed Saghayan et al. [333]	About 2 million tweets were gathered in this dataset from Twitter. The results demonstrate that machine translation has a 4% negative influence on binary accuracy classification and a 23% negative impact on multicast classification.	Used SVM classifier on extracted tweets.
Mohammad Mahdi Mahsuli et al. [176]	With an attention-based strategy and a corpus of 14,000 word pairings, the result reveals a performance improvement of more than 4.21 BLEU scores.	Deep learning, attention-based approach, Word2Vec, deep neural network for vector depiction.
Hadis Ghasemi et al. [148]	Chi-square outcome reveals the direction translation is not advantageous for qualitative Google translation.	Used Keshavarz (1999) fault analysis. Google translation mistake for Persian to English translation on both direction.
Mahdiyeh Alimohammadi et al. [127]	Corpus of Persian sign language. The technologies aided deaf and hearing youngsters in online communication.	Statistical machine translation, utilized Moses Machine Translation.
Hosein Azarbonyad et al. [88]	The results suggest that the phrase-based strategy is effective in cross-language information retrieval for English-Persian. The corpus improved by 106%, while the multilingual corpus improved by 7%.	Multiple translations for English-Persian cross-language information retrieval using a context-based technique.
Fahime Mohammadpour et al. [370]	The results reveal that the approach can modify numerous natural phrases using the appropriate Persian word order.	Eventually, bought hybrid machine Rule-based and corpus-based translation for English-Persian translation.
Abdol Hamid Pilevar et al. [56]	The result shows the Post-editing module is applicable for RBMT as a new domain. Used BLEU, TER metrics.	Used statistical post-editing of enhancing the quality of rule-based machine translation, Moses toolkit as statistical machine translation and SPE.
Mina Abbaszade et al. [295]	The outcome reveals the method is best and quicker in accuracy in contrast to the classical approach.	Used quantum natural language processing, Quantum long-short-memory (Q-LSTM) for translating of English-Persian.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Heshaam Faili [22]	Outcomes reveal 16% betterment of the NIST measure, and assessment of the WSD corpus of 860 aligned sentences, 81% of ambiguous words were accurate.	For autonomous learning of word meaning disambiguation in English-Persian machine translation, a decision tree was utilized.
Jan w. Amtrup et al. [10]	The approach is useful for locating and combining the correct morpheme for a word in a large corpus.	Finite State Transducers and Feature Description Structure improved the efficiency of morphology in MT.
Mehrdad Vasheghani Farahani. [377]	Seven distinct genre corpora. Two expert human translators. Human translation obtains 64.6% better than Bing and Google in the case of punctuation symbols. Both The machine translation got 54.5% and 53.6% accordingly.	Experiment on machine and human translation.
Ebrahim Ansari et al. [106]	Create a Persian-English bilingual corpus using Wikipedia as a distinguishing corpus. The results demonstrate that this strategy can significantly increase Persian-English translation performance.	The resemblance score between two phrases was computed using bidirectional translation, which was based on bidirectional novel translation. The length penalty score is used to improve the extracted corpus' accuracy.
Sanaz Kokabpeik et al. [369]	Created 12 questions in the form of a questionnaire, which were filled out by 100 people (males and females), the results reveal that the Google Translate system is not accurate enough to be utilized in academic areas.	Because of the chi-square assessment and bar chart, SPSS software was used.
Somayeh Bakhshaei et al. [67]	System can enhance the baseline system with low cost by just asking for translation of important sentences for system quality enhancement.	Used pool-based active learning for enhancing the Farsi-English MTs.
Mohammad Reza Mirsarraf et al. [98]	The results reveal that this technique assessment rate is strongly and positively connected with human judgment.	For a semantic assessment of phrases, used dependency-inspired semantics to evaluate a machine translation (MT) English-Persian dependency grammar.
Milad Maleki et al. [177]	Mezan corpus has one million translated phrases that may be matched with Google Translate text. The results suggest that human translation is still superior to machine translation in terms of quality.	Evaluate English-Persian MT in the situation of Ellipsis.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Benyamin Ahmadnia et al. [254]	The results reveal that the new technique is successful for English-Persian translation and has significantly higher quality than the English-Persian SMT model.	Improve English-Persian parallel neural network translation in low-resource environments.
Benyamin Ahmadnia et al. [297]	The results demonstrate that employing a large training corpus improves NMT performance. Named entities are frequently mistranslated by NMT. Instead of human translation, the technology may be relied on.	Attention-based NMT was used to predict and investigate translation performance. Improved the performance of Spanish-Persian neural MTs.
Hakiminejad Afshin et al. [142]	Google Translate cannot translate verb tenses from English to Persian, and passages are also unacceptable. GT made grammatical, aspect-dependent, passive, compound, lexical analysis, and discard mistakes.	In terms of verb tense in English-Persian MTs, compared Google Translation to human translation.
Reza Khanmohammadi et al. [321]	The results reveal that the system is capable of producing professional and non-professional Persian poetry and prose.	BERT is a transformer-based language with a novel neural machine translation technique.
Shahram Salami et al. [161]	The results demonstrate that the technique achieves a considerable improvement rate and outperforms the state-of-the-art phrase-based hierarchical model for translation from Persian, French, and Spanish to English lacking linguistic tools.	Phrase-boundary model for low-resource statistical machine translation, probabilistic synchronous context-free grammar, and non-terminal coarse-grained searching Word cluster portion of speech tags that have been trained For phrase pair alignment, a restricted pattern was used.
Nicola Bertoldi et al. [90]	Improving Text Normalization and creating a huge monolingual news corpus for Persian based on the baseline.	Enhance Persian-English and English-Persian over baseline supplied by the organizer of the FBK machine translation workshop in assessing campaign IWSLT 2013.
Razieh Esmailpour et al. [268]	In terms of performance, GT and QEAD are 60% identical. There is 40% dissimilarity. The system is performing better.	The Persian user question was translated using the Query Expansion Assistant database (QEAD) and a natural language processing technique.
Behrang QasemeiZadeh et al. [17]	A Farsi linguistic unity orthography, as well as an annotated corpus and lexicon, are being developed.	Using the MULTEXT-East Framework, create tools for writing systems and morphological analysis.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Jan W. Amtrup et al. [7]	Shiraz MT Persian-English,	A machine translation system, employs unification and typed feature structures.

2.2.6 Word Sense Disambiguation

S.M. Fakharahmad et al. [46] proposed a method for identifying word meaning disambiguation for a corpus of Persian translations of English utterances. The suggested technique took advantage of preprocessing. For extracting the knowledge about word duplication. The exact notion of inductive management for ambiguous terms was discovered in the second stage using a forward-changing methodology. When compared to other systems, the new system's experiments demonstrated increased accuracy.

M.nameh et al. [54] the word meaning ambiguity based on cosine similarity might be resolved using supervised machine learning. The TWA dataset is used to construct the suggested system. Finding the co-occurrence of a term that often appears in the text and numerous other words that include ambiguous words comes first. The second stage involved the author using a 10-fold cross-validation method to separate the training from the test portions of a dataset for context similarity categorization. In the majority of situations, the outcome is satisfactory.

Heshaam Faili et al. [146] explains a method for finding grammatical and spelling errors in Persian literature. Based on Vafa Spell-Checker, word sense disambiguation problems in written Persian in the actual world.

Behzad Moradi et al. [243] provided a method for unsupervised word embedding-based word sense disambiguation. The word2Vec approach, which is well-known for English word embedding, is used in the suggested system for training English. The goal of this strategy is to disambiguate Persian words by using their English translations as surrounding words. The final translation is chosen based on the related word with the greatest score. Other languages can likewise use the current method. A test dataset manually created for the recommended approach was used to determine the accuracy of the present system.

Masoud Narouei et al. [135] a method based on supervised and unsupervised techniques for word sense disambiguation for sentences in sequential patterns. The suggested approach uses variable-length size, n-grams, to dynamically locate pertinent words regardless of where they appear in the sentences. Even though the author employed monolingual characteristics, experiments reveal that the present strategy improves the precision of the multilingual approach by more than 4%. SePaS accomplishes an overall accuracy of 67.2, higher than MFS and lower than IMS.

Mohamadreza mahmoodvand et al. [175] presented a method for figuring out the meaning of Persian words that was semi-supervised. For word sense disambiguation, the suggested system used a hybrid machine learning technique with a light supervision mechanism. In the assessment phase, a corpus that is added from the data of news agencies is used, which demonstrates. For understanding the intent of 5,368 documents. With accomplishing an accuracy of 88% in recall and 95% in precession, the overall accuracy of 93%.

Mortaza Montazery et al. [53] Based on an unsupervised learning technique, Persian WordNet was produced. Expectation-Maximization (EM), a corpus, and Princeton WordNet 3.0 (PWN) were used in the proposed method as an unsupervised learning data source. The current method can be used to WordNet networks for all languages. Eventually, the current system accomplishes higher than 86% precision in transforming 7,109 from, 11076 different words and 9,427 different synsets.

Mohamadreza mahmoodvand et al. [134] provided a technique for figuring up Persian word meanings. based on the web-crawler corpus extraction approach and applied to the Hamshahri corpus. The crawling robot used in the suggested method collects the intriguing ambiguous words; the word is added to the Persian dataset as an ambiguous word for extracting sentences that exactly include the word. This strategy is workable and provides enough words for the use of supervised and semi-supervised approaches.

Bahareh Sarrafzadeh [59]proposed a cross-lingual word meaning disambiguation for Persian. The present technique employs FarsNet and a parallel corpus to automatically construct massive, sense-tagged corpora. The experiment's findings demonstrated that the cross-lingual approach is more effective than a monolingual approach, and that its performance is equivalent to that of an English sense tagger.

Saeed Rahmani et al. [328] introduces a new unsupervised learning method for the Persian word meaning disambiguation that uses graph-based context adoption. The representation of ambiguous words in the suggested technique is a sub-graph loaded from co-occurrences of word graph-based corpus. Relations and senses are placed onto a graph to determine the true meaning of an ambiguous word. Selection of word senses using a monolingual corpus from the context graph with the highest degree of similarity. The recommended technique is compared against existing unsupervised and supervised WSD algorithms in experiments on datasets in English and Persian.

Abdoreza Rezapour et al. [122] based on weighted lexical characteristics, word-to-sense disambiguation for Persian was presented. The K-Nearest Neighbor (K-NN) algorithm, a supervised learning technique on tagged-sense corpus, served as the foundation for the suggested method. Extract the words that appear together often in the corpus and words that are close to ambiguous terms in the first step. The corpus for K-NN classification was better divided into tests and training with the usage of a cross-validation technique. The recommended system outperforms other recent and previous methods in terms of outcomes.

Navid Rekabsz, et al. [158] produced a cross-linguistic, stranded test set for word sense disambiguation between English and Persian. The SemEval-2013 CL-WSD job for the Persian language was updated using the current methodology from the SemEval-2013 CL-WSD task.

Noushin Riahi et al. [159] provided a method to enhance the extraction of collocation methods for Persian words meaning disambiguation using an untagged corpus. With a small labeled corpus and a large untagged corpus, the proposed method seeks to enhance the supervised decision list algorithm. The accuracy of a supervised system that was trained on a small tagged corpus is comparable to that of a larger untagged corpus. The outcome is great for both the large untagged corpus and the tiny tagged corpus. The system exhibits improvement for different homographs from 1 to 3% process.

Elham Alayiaboozar [190] provided a machine learning approach based on the Lesk algorithm and POS tagging for figuring out the meaning of Persian words. In the suggested method, POS tagging is done initially, followed by POS disambiguation in Persian adjectives, nouns, and homographs. The stage Lesk algorithm is completed for word sense disambiguation at this point. By limiting the filter to just linked lexically existent words, the recommended solution speeds up the WSD technique.

Bahareh Sarrafzadeh [60] to aid semi-supervised and supervised WSD systems, a sense-tagged corpus was created. The method used a cross-lingual WSD technique with word sense disambiguation in parallel corpus to improve the transfer sense stage by incorporating a direct sense tagging phase. On the tagged corpus, which has a minimal error rate of 9%, the outcome was encouraging.

Mahmood Soltani et al. [44] offered a statistically based strategy for resolving ambiguity in Persian word senses. The suggested graph-based word sense disambiguation method is used to determine how related words are semantically and to resolve word ambiguity. The experiment demonstrates that the performance of the English-Persian translation was substantial.

Noushin Riahi et al. [83] gave a strategy based on a statistical approach and a semi-supervised approach for resolving ambiguity in Persian word meanings. Due to the lack of a huge labeled corpus, the author's suggested solution uses a semi-supervised approach. The tiny tagged corpus responds well to the semi-supervised technique. By applying coarse-grained pre-training, the corpus experiment system exhibits high accuracy when compared to conventional judgments.

Elham Alayizboozar et al. [221] recommended a method for Persian word sense disambiguation based on a rule-based approach concentrating on POS tag disambiguation. The suggested system recognizes the collection of adjectives, and noun homographs ending with, (<ی>) ya for correct POS tagging in Persian.

F.Jani et al. [76] used the Hamshahri corpus, a typical Persian corpus, to disambiguate the meaning of Persian words using supervised and unsupervised methods. The present technique managed to attain an accuracy of 97%.

Nasrin Taghizadeh et al. [164] using cross-lingual word sense disambiguation, an automated Persian WordNet was created. In the recommended method, the Expectation-Maximization algorithm is applied for creating a high-quality WordNet utilizing a monolingual corpus and parallel dictionary for under-resourced languages. WordNet's findings and the Persian language were used in the experiment. The induced WordNet obtains recall scores of 35% and precision rates of 90%.

Hossein Rouhizadeh et al. [330] Persian word sense disambiguation for all terms was supplied in an annotated corpus. The suggested system is based on SBU-WSD-Corpus, which contains papers from 19 different Persian domains, including science, the arts, sports, and others. Using the corpus's manual annotations (FarsNet).

Hossein Rouhizadeh et al. [329] produced a Persian Bag-of-Words corpus called PerSemCor, which contains an annotated sense corpus. The experiment's corpus displays positive word meaning disambiguation findings for Persian.

Hossein Rouhizadeh et al. [284] a novel knowledge-based methodology for Persian word sense disambiguation has been suggested. Working with a trained model and LDA. Calculating word similarity using the Persian WordNet. In comparison to earlier knowledge-based techniques, the author's implementation of the proposed methodology on an all-word word sense disambiguation corpus yields encouraging results.

Maryam Haghollahi et al. [47] added additional new relations to FarsNet using a semi-supervised method for leveraging in-word meaning disambiguation. The recommended method extracts semantic keywords from textual annotations. The experiment demonstrates that the existing approach, which was evaluated for the Persian language as well as the proposed WSD and the proposed system, enhanced accuracy. The system is flexible for other languages, such as English.

Navid Rekabsaz et al. [182] provided a strategy for the application of English into Persian utilizing an unsupervised method based on word sense disambiguation across many languages. The suggested language utilizes semantic similarity and word embedding of a word in context to work on the target language as a low-resource language. In the experiment, the author compared the recommended system to Co-Graph, a cutting-edge unsupervised system, to evaluate it. The best performance out of five was achieved with the present approach. A present method is a suitable option for languages without parallel corpora or lack of information [354].

Table 2.10: Persian Word Sense Disambiguation

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
S.M. Fakharahmad et al. [46]	A corpus of English sentences that have been translated into Persian. When compared to counterparts, the result shows an improvement in precision.	Preprocessing is done to extract knowledge about duplicate words in, a forward-changing strategy to discover the precise sense of inductive management for ambiguous words.
M.nameh et al. [54]	TWA sense tagged dataset. The results reveal that the method is satisfactory in the vast majority of circumstances.	In a dataset of context similarity classification, the supervised machine learning for WSD. Cosine similarity, and utilized a 10-fold cross-validation strategy for separating training and test portions.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Heshaam Faili et al. [146]	Presents a method for detecting spelling and grammatical errors in Persian text.	Based on Vafa Spell-Checker, the word meaning disambiguation in the Persian language is similar to mistakes in the actual world.
Behzad Moradi et al. [243]	The output presents a manually produced test dataset describing the current system's accuracy.	Word2Vec technique for unsupervised word embedding for Persian WSD.
Masoud Narouei et al. [135]	The experiment made use of the TWA dataset, the SenseEval3 English lexical sample, and the SemEval-2007 word sense induction dataset. In the trial, the multilingual system improved by 4%, with 67.2 accuracies greater than MFS and lower than IMS.	Supervised and unsupervised approaches for WSD in Persian using phrases in a sequential pattern and variable-length size n-grams.
Mohamadreza mahmoodvand et al. [175]	In a corpus compiled from news agencies, the recall is 88%, the precision is 95%, and the total accuracy is 93%.	A mix of machine learning approach and minimal supervision method for disambiguating word senses.
Mortaza Montazery et al. [53]	The present system achieves greater than 86% precision in converting 7,109 from, 11,076 distinct words, and 9,427 different synsets using Princeton WordNet 3.0, Expectation-Maximization.	Make a Persian WordNet using unsupervised learning.
Mohamadreza mahmoodvand et al. [134]	The outcome of the Hamshahri corpus reveals that enough words may be combined to use supervised and semi-supervised techniques.	Method for extracting corpora from web crawlers for word sense disambiguation. In the field of WSD, for obtaining standard corpora for the Persian language.
Bahareh Sarrafzadeh. [59]	The performance of the parallel corpus and FarsNet is the same as that of the used English sense tagger.	Persian WSD cross-lingual technique

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Saeed Rahmani et al. [328]	The suggested technique is described in comparison to current unsupervised and supervised WSD techniques using a monolingual corpus and experiments on English and Persian datasets.	Adoption of graph-based context as a novel unsupervised technique
Abdoreza Rezapour et al. [122]	In comparison to the traditional technique used in Persian WSD, the new system achieves an excellent result using the tagged-sense corpus.	Weighted lexical features, K-NN method, and cross-validation for Persian WSD
Navid Rekabsz, et al. [158]	Persian WSD standard corpus was created.	The current approach altered in the SemEval2013 CL-WSD workshop.
Noushin Riahi et al. [159]	Large untagged corpora and small tagged corpora. When comparing large untagged corpora to tiny tagged corpora, the large corpus performs well. The technique improves the procedure for unique homographs by 1 to 3 percent.	The supervised algorithm was utilized to improve the decision list algorithm and the extraction of collection techniques for Persian WSD.
Elham Alayiaboozar presented. [190]	The suggested approach improves the speed of the WSD technique by filtering only related paraphrases from the dictionary.	The Lesk method is an unsupervised algorithm with POS tagging that is utilized for Persian WSD.
Bahareh Sarrafzadeh. [60]	To improve the transfer sense stage by incorporating a direct sensation tagging step. On the tagged corpus, the results were encouraging. It has a 9% error rate.	Developed a sense-tagged corpus to aid semi-supervised and supervised WSD systems for the Persian, automated method.
Mahmood Soltani et al. [44]	The investigation on English-Persian translation performance yielded substantial results.	Statistical technique for Persian WSD based on a graph-based algorithm.
Noushin Riahi et al. [83]	The new strategy outperforms a traditional choice that employs a coarse-grained method in terms of accuracy.	A semi-supervised technique for Persian WSD because of the paucity of big labeled corpora.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Elham Alayizboozar et al. [221]	The suggested system determines the collection of adjective and noun homographs that end with, (<ی>) for accurate POS tagging in Persian.	Rule-based approach for Persian WSD.
F.Jani et al. [76]	The present technique achieved 97 percent accuracy on the Hamshahri corpus.	For Persian WSD, supervised and unsupervised techniques were used.
Nasrin Taghizadeh et al. [164]	With a parallel lexicon and a monolingual corpus, the induced WordNet achieved an accuracy score of 90% and a recall score of 35%.	Built an automatic Persian WordNet for Persian WSD using a cross-lingual, maximization algorithm.
Hossein Rouhizadeh et al. [330]	SBU-WSD-Corpus, which had 19 Persian domain distinct papers.	Create an annotated Persian corpus for WSD; the corpus is manually annotated using FarsNet.
Hossein Rouhizadeh et al. [329]	The results demonstrate that the Persian supervised WSD performed well.	PerSemCor is a Persian Bag-of-word corpus containing annotated sense corpus.
Hossein Rouhizadeh et al. [284]	Persian WordNet used the current technique for all-word, WSD corpus, with good results when compared to other knowledge-based approaches.	A knowledge-based strategy for Persian WSD utilizing the LDA pre-trained model.
aryam Haghollahi et al. [47]	The outcome demonstrates strategies for improving the accuracy of the proposed WSD for Persian.	FarsNet was enhanced using a semi-supervised technique.
Navid Rekabsaz et al. [182]	Out of the top five systems, the system performs well.	Unsupervised technique based on cross-lingual WSD for English-Persian application using word embedding and semantic similarity.

2.2.7 Sentiment Analysis

Mursal Dawodi et al. [266] demonstrate the feature extraction and machine learning strategies for Dari sentiment analysis. Building functional resources are significant, especially for languages with few resources like

Dari. Additionally, Naive Bayes, Multinomial Naive Bayes, Support Vector Machine, Neural Network, Random Forest, Nearest Neighbor, Decision Tree, and Logistic Regression were used in a case of sentiment analysis in the Dari language. On a dataset of more than 8,000 reviews that had been manually classified as positive and negative, they used a variety of machine learning techniques. they have utilized a variety of feature extraction approaches for the preparation of the data to arrange the text characteristics in an improper numeric format and choose the best approach for Dari sentiment analysis. Three distinct feature extraction techniques—TF IDF, (vector of bigram) Bag-of-Word model, and the Bag-of-word model (vector of unigram) —were used throughout experimentation. Additionally, accuracy, precision, recall, and F-score are four assessment measures that they used to evaluate each process. The accuracy they achieved by these approaches is equal to 91 percent, 89 percent, and 89 percent, respectively, for the greatest performance they obtained using Multinomial Naive Bayes, Support Vector Machine, and Logistic Regression. Additionally, the bag of the word (vector of bigram) showed greater performance when compared to other feature extraction techniques. In comparison to employing the Bag of Word (vector of bigram) technique, the Multinomial Naive Bayes Method has demonstrated the best performance. Reza Akhoundzade et al. [255] the unsupervised method-based technique for sentiment analysis in Persian. The suggested method made use of word-embedding models, rule-based systems, k-mean clustering, neural networks, and rule-based systems. The efficiency of the recommended system was perfect, with the F1 measure yielding an aspect word of 0.766 when compared to the supervised technique.

Mohammad Ehsan Basiri et al. [168] assessed the four lexicons for sentiment analysis in Persian, 1: NRC; 2: Adjectives; 3: CNRC; and 4: SentiStrength. The examination demonstrates that further pre- and post-processing is required for excellent performance, and that straight translation is insufficient. Additionally, adjectives perform better than NRC and CNRC. This study set is intended to highlight the shortcomings of Persian sentiment analysis and to improve its application through the use of more accurate lexicons and lexicon-based sentiment analysis. For research, it makes sense to use lexicon-based, machine learning-based methods.

Kia Dashtipour et al. [305] Describe a strategy for supervised Persian sentiment analysis. For locating every polarity in the Persian movie review dataset, the suggested approach employs TF-IDF with the unigram feature. The 1,000 good and 1,000 negative reviews in the present dataset were gathered from www.caffecinema.com and www.cinematicket.org. The results of the trial demonstrate that TF-IDF was more effective than frequency. The IDF transition point was very beneficial in contrast to Frequency and TF-IDF. For future work, the author proposes developing a multilingual sentiment analysis for Persian, Arabic, and English that can recognize all polarities.

Mohammad Ehsan Basiri et al. [260] demonstrate the consequence of the aggregation procedure of Persian reviews sentiment classification. The suggested system's results on four datasets related to Persian reviews demonstrate that aggregating reviews at the level of categorization can improve rating classification, but that categorization of polarity is unaffected by this approach.

Monire Shirghasemi et al. [338] demonstrates the impact of active learning algorithms on cross-lingual person sentiment tasks. The suggested system chooses the most informative sample using active learning and then presents it to the user for labeling. As is well-known, labeling data comes at a significant time and expense. Cross-lingual word embedding also employs active learning. Utilizing the Active Learning method, they were able to significantly reduce the amount of labeled data required for this problem. These techniques let our classifier comprehend samples better and work more efficiently.

Mohammad Dehghani et al. [363] provided a method for analyzing political tweets' sentiment in Persian using machine learning techniques. The suggested method makes use of Gradient-Boosting, Random Forest, Support Vector Machines, and Decision Trees to predict the polarity of tweets on politics. Random forest outperformed the other approaches with a performance of 86%.

Ali Derakhshan et al. [230] social media for Persian stocks allows people to anticipate price movement using sentiment analysis. The suggested approach makes use of topic modeling approaches that leverage part-of-speech. The LDA-POS approach produced results with 56.24 and 55.33 percent of the whole of the English and Persian language corpus, respectively. The current methodology performed poorly for the Persian language, but was better for the English dataset. Persian and English were equivalent to the LDA-POS method's output. The Persian dataset included five equities, price changes, and user ratings.

Ayoub Bagheri et al. [89] worked on the Persian language customer review. It is important to address issues with word spacing, declension suffixes, intra-term space, and colloquial terms in the text. The suggested approach is based on the naive Bayes learning algorithm and lemmatization for classification. Using a mutual information strategy, the optimum combination of characteristics may be extracted from the initial features that are extracted. The experiment relied on manually gathered mobile phone evaluations to assess performance, and the outcomes were quite effective and helpful for the recommended strategy.

Behnam Roshanfekr et al. [184] provided a method for sentiment analysis of Persian text using deep learning. An experiment demonstrates that deep learning techniques function well and have a higher F measure than Naive Bayes SVM (NBSVM). In this study, the author developed a dataset for a Persian-language electronic product review that is also utilized for evaluation. The many writing styles in Persian literature present the suggested model with considerable complexity.

Aria Naseri Krimvand et al. [320] provided a Deep Learning-based method for sentiment analysis in Persian. The suggested approach is tested on the MPerInst Instagram dataset, which consists of 512 image pairings and associated Persian-language comments. They employed a 2-dimensional convolutional neural network (2CNN) for image processing and a bi-GRU for text processing. The result of the present strategy is that the system accurately identifies polarity when both strategies are used. F1 measure rises by 23 and 0.24, which is better than utilizing just text and images to convey ideas. The new model outperforms 11 comparable deep fusion models by 11% and 0.1.

Mohammad Karrabi et al. [273] provided a method for Persian sentiment analysis based on attention-based LSTM networks and word embedding. Taaghche and Filimo, two Persian corpora, were subject to the suggested system. An experiment demonstrates the effectiveness of employing quick text-based vector representations of words for sentiment analysis. In Persian sentiment analysis, using an attention model can improve the deep neural Network (DNN) prediction accuracy.

Mohammad Ehsan Basiri et al. [167] offer a SPerSent dataset for sentiment analysis at the sentence level in Persian. Incorporate a fresh Persian lexicon into CNRC as well. 150,000 phrases with two labels are included in SPerSent; the first label denotes sentence polarity and the second label displays a five-star rating. The Naive Bayes approach was employed to assess SPerSent.

Souroush Karimi et al. [232] Describe a method for sentiment analysis in Persian based on deep learning and BERT's pretraining language representations. A dataset with a total of 200,761 customer reviews for electrical devices was used by the author. BERT works better than skip-gram, LSTM, or CNN, according to an experiment.

Fatemeh Amiri et al. [128] gave a lexicon-based method for sentiment analysis in Persian. The Gazetteer Persian emotion lexicon was employed in the suggested system. According to an experiment, the expanded GATE pipeline exhibits overall accuracy of more than 69%.

Mahsa Heidari et al. [270] worked with Instagram data which deleted close to 111,000 comments. Nearly 9,000 of them were labeled using crowdsourcing techniques. It is verified by using the Word2Vec dataset. On the Kaggle website, a recent dataset with the name of Insta-Text is publicly accessible.

Reza Akhoundzade et al. [220] provided a methodology based on the unsupervised technique for sentiment analysis in Persian. The dataset used in the present methodology contains complete, unfiltered assessments of computers, tablets, and phones that were gathered from the Digikala website. The suggested system makes use of a rule-based approach together with the Word2Vec neural network model. Additionally, a cutting-edge technique was applied to increase the vocabulary of Persian feelings. The system obtains an F1 measure of 0.58 for the experiment's present approach.

Mohammad Aliramezani et al. [256] to get over the lack of labeled datasets for Persian, a cross-lingual approach to sentiment analysis in Persian was developed. Because English is a language with plenty of resources, the author's suggested technique employed English as a training language. Word embedding alignment is being brought out to be used as a feature vector. To test align word embedding for English-Persian, the author trained in English and Persian using Fast text and GloVe. The method uses Persian Snapp Food as a test dataset and Amazon fine food reviews as a training dataset. The projected system accomplished 78.1% F1 measure with Persian test data.

Kia Dashtipour et al. [145] created a PerSent Persian sentiment analysis lexicon that is freely accessible to academics. There are 1,500 words in the current vocabulary, together with polarity ratings and part of speech tags. by analyzing the present vocabulary with a machine-learning technique.

Sajjad Shumaly et al. [339] describe a method for sentiment analysis in Persian based on convolutional neural networks and fast text embedding. The authors employed CNN and Fast Text techniques to get over the aforementioned issue because there are no preprocessing tools for Persian. The largest e-commerce website in Iran, Digikala, provides statistics for the system's implementation that comprise roughly 3 million reviews. The author employed a pseudo-labeling technique to improve the outcome due to the dearth of labels in a significant portion of the dataset. The system attained an F1 measure of 0.956 and an accuracy of 0.996.

Mohammad Saree et al. [103] give a method for feature selection in sentiment analysis in Persian. The challenge approach is suggested as a way to get around conversional words, different word spacing, and declension suffixes. Utilizing the Naive Bayes technique for categorization. The author employed several mobile phone reviews to evaluate the existing strategy, and the outcome was quite positive.

Faranak Ebrahimi Raashed et al. [172] based on the supervised technique, create a vocabulary for Persian sentiment categorization. The present approach is corpus-based and tries to create a Persian emotion dictionary utilizing part-of-speech tags and mutual information to identify word polarity. The proposed approach ultimately achieves an accuracy of 80%.

Hamoon Jafarian et al. [317] used BERT to improve Aspect-Based Sentiment Analysis for Persian. A supplemental natural language inference sentence and sentence pair input on an ABSA task is used in the suggested technique, which uses ParsBERT and a pretrained model. The technology was put in place to improve the performance of the Persian Pars-ABSA and ABSA databases. According to the most recent experiments, the system improves ABSA performance with 91-percent accuracy, which is 5.5 percent better than the Pars ABSA dataset.

Kia Dashtipour et al. [303] provide a Deep Learning method for analyzing Persian emotion in movie reviews. In the trial, stacked bidirectional-LSTM, using a movie dataset, obtained an accuracy of 95.61%, whereas 2D-CNN, using a hotel dataset, achieved an accuracy of 89.76%.

Kia Dashtipour et al. [306] present ensemble classifiers for Persian Sentiment Analysis utilizing shallow and deep learning to boost the effectiveness of cutting-edge methods. The dataset utilized for this method's analysis of hotel reviews has 3,000 total reviews, 1,500 favorable reviews, and 1,500 negative reviews. The experiment shows that the present method yields an accuracy of over 79.68%.

Mohammad Heydari et al. [315] present a hybrid Deep Learning-based approach to sentiment analysis for Persian. The online e-commerce website of Digikala has 100,000 client reviews. The present strategy scores 78.3 percent on the F1 test for positive, negative, and neutral polarities.

Kia Dashtipour et al. [349] increase the number of colloquial expressions with their polarity in the PerSent Persian sentiment lexicon to about 1,000. The improved version is helpful for categorizing Persian text as well as for phrase recognition. Contrary to PerSent, which is lexicon-based, several studies demonstrate the usefulness of the expanded form of the lexicon for sentiment analysis.

Kia Dashtipour et al. [170] provided a method for determining the polarity of emotion in Persian movie reviews using part-of-speech tags and N-grams. On the Persian lexicon PerSent, the feature polarity is fixed. The author employed the SVM classifier to demonstrate in the experiment assessment that integrating unigram, bigram, and Trigram features will result in the greatest performance.

Rouzbeh Ghasemi et al. [351] for overcoming the lack of reliable training data for low-resource languages like Persian in natural language processing technologies. In this study, the author presented a cross-lingual deep learning architecture to make use of the English language's abundant training data. An experiment was conducted using the Persian Digikala dataset and the Amazon dataset, with the present methodology boosting Persian sentiment analysis by 22%. For languages that translate from source to target language, the bilingual dictionary is reliable for cross-lingual embedding.

Batoul Botshekanan Dehkordi et al. [195] give a method for determining polarity in phrases that contain both positive and negative words. To determine the sentences' scores, the absolute value of the CNRC lexicon was used. The author used the Digikala corpus, which comprises 151,229 sentences. Each sentence was given a score between 1 and 5, with 1 and 2 for negative statements, 3 and 4 for positive sentences, and 3 for neutral sentences. An experiment demonstrates that the lexicon-based technique for two polarity detections and score prediction issues may be improved through bipolar words.

Rahim Dehkharghani [229] built a Persian polarity lexicon using a translation-based methodology for creating polarity resources. The current SentiFars Persian lexicon is effective, according to experimental evaluation.

Mohammad Ehsan Basiri et al. [192] improved Persian sentiment analysis by lexicon refinement and overcame the lack of resources for sentiment analysis in Persian. In the suggested approach, machine learning algorithms are trained using PerLex words. The PerView dataset was utilized in the evaluation. In comparison to the present CNRC, Adjectives, SentiStrength, PerSent, and LexiPers lexicon, the result demonstrates that PerLex performs well. The execution time of the current method was cut down by using PerLex instead of the lexicon that was described earlier.

Milad Vazan et al. [344] tried to resolve the difficulties of aspect-category-detection ACD and aspect category polarity ACP. On a Persian reviews dataset collected from movie ticket websites, which includes 2,200 samples from 14 categories, a performance experiment was examined. The system was assessed using label-based and example-based criteria, and the results suggest that CNN and GRU models are to be prioritized above LSTM and Bi-LSTM models.

Kia Dashtipour et al. [304] offer context-aware multimodal sentiment analysis, a novel analytical framework that is used to perform multimodal sentiment analysis. The suggested approach made use of decision-level (late) and feature-level (early) fusion techniques to effectively combine cross-model data from a Persian multimodal dataset with more than 800 expressions. The results of the experiment demonstrate that contextually mixing multimodal variables, such as textual, audio, and visual features, results in accuracy that is 91.39 percent greater than unimodal, which is 89.24 percent.

Nazanian Sabri et al. [331] provide a code-mixed dataset for Persian-English that contains, 3,640 acquired tweets from the Twitter API, each of which is tagged with the polarity that corresponds to it. The author makes use of a neural classification model to learn polarity scores. He utilized Yandex and dictionary-based to convert cod-mixed words into a text. The author employed a pre-trained word embedding BERT to represent the data. In the end, the system had a data accuracy of 66.17 percent and an F1 of 63.66.

Seyed Arad Ashrafi Asli et al. [257] present MIrasOpinion, a sizable Persian sentiment analysis dataset that was assessed using LSTM and CNN and achieved an F1 score of 80% accuracy. The author also introduces a brand-new active learning technique called LDA sampling for text-based categorization that groups the representation of a document into different categories. Superior results as compared to other methods.

Rahim Dehkharghani et al. [267] proposed a deep neural network and classifier-based approach to sentiment analysis of Persian. The suggested approach uses multiple granularity levels, including aspect, word, phrase, sentences, and document levels, to address language issues including intensification, negation, and granularity. Additionally, applicable to various languages, is the recommended system. The assessment stage demonstrates how effective this strategy is in comparison to other strategies.

Javed PourMoustafa Roshan sharami et al. [287] present an LSTM and CNN-based deep learning method for sentiment analysis of Persian text. The suggested method seeks to extract opinions from the text at the sentence level. Additionally, three data augmentation methods for the Persian sentiment analysis corpus were recommended. The proposed approach was effective and in line with the objective of the ongoing study. In the experiment, word embedding and neural word embedding methods define our data augmentation methods.

Behnam Sabeti et al. [248] developed the Persian sentiment lexicon-based ontology LexiPers. A new graph-based method for seed selection is described, together with a K-nearest neighbor used in classification in the suggested approach. The sentiment analysis algorithm displays f-measure and believable accuracy.

Kia Dashtipour et al. [263] give a hybrid strategy for Persian sentiment analysis based on integrity dependence grammar-based rules and a Deep Neural Network. The Product Reviews dataset, which was used by the author, includes 1,500 of which are favorable, and 1,500 which are negative, and are collected from the Digikala website. He also made use of the Hotel Evaluations dataset, which compiles 1,800 favorable and 1,800 unfavorable reviews are collected from hotel booking websites. For improving polarity recognition in sentiment analysis, the suggested model combined deep learning with integrated linguistic rules. Using the hotel reviews dataset, the experiment showed that the present technique beat state-of-the-art approaches including logistic regression, support vector machine, DNN classifier LSTM, and CNN with the biggest margins of 10–16% and 3–4%, respectively.

Taha Shangipour ataie et al. [250] present Pars-ABSA[35], an aspect-based dataset for sentiment analysis in Persian. Three Persian native speakers tested and certified the present dataset. The current dataset has 5,602 distinct reviews with 5,114 positive, 3,061 negative, and 1,827 neutral evaluations. The experiment's current dataset produces a remarkable outcome that is comparable to English State-of-the-art techniques.

Kia Dashtipour et al. [196] propose a Deep Learning-based method for sentiment analysis in Persian. The suggested technique implemented a Persian movie review dataset using deep convolutional neural networks and

[35] https://github.com/Titowak/Pars-ABSA

deep autoencoders. An experiment demonstrates that the performance is improved when compared to cutting-edge MLP.

Pedram Hossenini et al. [200] officially announce SentiPers[36], a dataset for Persian. More than 26,000 sentences are included in the aforementioned corpus.

Mohammad Bagher Dastgheib et al. [264] propose a technique for sentiment analysis of Persian utilizing Structural Correspondence Learning (SCL) and a Convolutional Neural Network (CNN). By using a hybrid technique that produces positive results for sentiment classification for two domains more than 10%, the suggested method performs effectively.

Ayoub Bagheri et al. [107] give a solution to the problems caused by the word spacing of declension suffixes and the use of various colloquial words. The suggested model relies on lemmatization and feature selection to function. employed a Naïve Bayes algorithm for categorization as well. Cell phone reviews are implemented by the system and are manually gathered. The outcome demonstrates how efficient and successful the existing strategy is.

Zahra Bokaee et al. [246] included hybrid deep learning Persian sentiment analysis using CNN-LSTM. The suggested solution employed LSTM for learning long-term dependencies and CNN for feature extraction. The system used both models and attained an accuracy of 85%. Two Persian datasets from Twitter and the well-known e-commerce site Digikala are used to test the recommended approach. Databases with the labels “positive”, “negative”, and “neutral”. The dataset is particularly helpful for Persian speakers.

Mohammad Ehsan Basiri et al. [108] gathered words according to their polarity of emotion into a Persian Polarity Lexicon. After identifying the issues of informal language, word spacing, stemming, and misspelling, the author looks for an effective solution. To categorize the polarity and analyze the system’s performance, use online cell phone reviews. The outcome demonstrates that the system outperforms cutting-edge supervised machine learning techniques [354].

[36] https://github.com/phosseini/SentiPers

Table 2.11: Persian Sentiment Analysis

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mursal Dawodi et al. [266]	The dataset of over 8,000 reviews that has been carefully tagged as positive and negative are equivalent to 91%, 89%, and 89%, respectively.	Naive Bayes, Multinomial Naive Bayes, Support Vector Machine, Neural Network, Random Forest, Nearest Neighbor, Decision Tree, and Logistic Regression utilized in method, TF-IDF, Bag-of-Word model (vector of unigram), and Bag-of-Word model (vector of bigram) used for feature extraction methods for preprocessing the data
Reza Akhoundzade et al. [255]	When compared to the supervised technique, the outcome was quite efficient, with a high F1 score for coming out as an aspect word of 0.766.	For Persian SA, an unsupervised technique, K-mean clustering, neural network, rule-based system, and word-embedding were used.
Mohammad Ehsan Basiri et al. [168]	Machine learning based on lexicons is an excellent approach for study, according to the evaluation.	NRC, adjective, CNRC, and SentiStrength were evaluated as lexicon for Persian sentiment analysis.
Kia Dahtipour et al. [305]	A collection of 1,000 negative and 1,000 favorable movie reviews. The results demonstrate that TF-IDF was superior to frequency, while IDF transition point was superior to both TF-IDF and frequency.	Preprocessing occurs following normalization and the removal of stop-words. unigram, bigram, and trigram features are extracted. TF-IDF was used with supervised algorithms and unigram features.
Mohammad Ehsan Basiri et al. [260]	The findings, which were implemented on four datasets of Persian reviews, demonstrate that aggregate review-level can improve rating categorization, but the strategy is ineffective for polarity classification.	Explored the impact of the aggregation approach of Persian reviews sentiment categorization.
Monire Shirghasemi et al. [338]	The results suggest that the strategy is beneficial in terms of interpreting samples and labeling data more efficiently.	Active learning in a cross-lingual model for sentiment analysis in Persian.
Mohammad dehghani et al. [363]	The random forest strategy performed the best in the political tweet dataset, with an 86% performance.	Gradient boosting, random forest, support vector machine, and decision tree are machine learning techniques for Persian SA.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Ali Derakhshan et al. [230]	A large corpus of English-Persian results in 56.24% and 55.33%, respectively. The present method worked well for the English dataset, but not so well for the Persian dataset.	LDA-POS for sentiment analysis of Persian stock price movement prediction on social media.
Ayoub Bageri et al. [89]	Dataset of cell phone reviews, The outcome demonstrates the efficacy and utility of the offered schemes.	For classification, lemmatization, and a naive Bayes learning method were used. In addition, to overcome the word spacing difficulty, a mutual information mechanism was adopted.
Behnam Roshanfekr et al. [184]	In the electronic product review dataset, the technique outperforms NBSVM in terms of performance and F-measure.	Deep learning for Persian sentiment analysis.
Aria Naseri Krimvand et al. [320]	Instagram dataset from MPerInst. The approach identifies polarity accuracy and F1 score 23 and at 0.24 better than using only picture and text distinctively, present model by 11% and 0.1 performance better than 11 similar deep fusion models.	For Persian SA, deep learning employed bi-GRU for text processing and 2CNN for image processing.
Mohammad Karrabi et al. [273]	Using an attention model on the Taaghche and Filimo datasets can improve DNN prediction accuracy in Persian SA.	Attention-Based LSTM network based on word embedding for Persian SA.
Mohammad Ehsan Basiri et al. [167]	Dataset New vocabulary for Persian SA and SPerSent, which comprises 150,000 phrases with two labels, the first indicating polarity and the second indicating a five-star rating.	A Naive Bayes system is used to evaluate the approach.
Souroush Karimi et al. [232]	A customer review dataset that contains 200,761 reviews of electronic products. When compared to skip-gram, LSTM, or CNN, the approach excels.	BERT, deep learning and pretraining linguistic representations
Fatemeh Amiri et al. [128]	Using the Gazetteer Persian sentiment lexicon, the method produces an overall accuracy of 69%.	Lexicon based for Persian SA.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mahsa Heidari et al. [270]	The Insta-Text dataset is accessible free on the Kaggle website,	Word2Vec for authorizing the dataset.
Reza Akhoundzade et al. [220]	A dataset of unfiltered smartphone, tablet, and laptop reviews. For the current approach, an F1 measure of 0.58 is obtained.	For Persian SA, use an unsupervised technique and a Word2Vec neural network.
Mohammad Aliramezani et al. [256]	Amazon fine food reviews were used as a training dataset, while Persian Snapp food was used as a test dataset. Using Persian test data, the suggested technique earned a 78.1% F1 score.	A cross-lingual approach for Persian SA was developed, and GloVe was utilized to test align word embedding for English-Persian.
Kia Dashtipour et al. [145]	PerSent Persian SA lexicon.	Machine Learning algorithm for assessment of current lexicon.
Sajjad Shumaly et al. [339]	The method achieves an accuracy of 0.996 and an F1 score of 0.956 in the dataset of around 3 million reviews.	CNN, FastText embedding Because of a lack of preprocessing tools for the Persian language.
Mohammad Saree et al. [103]	The outcome of numerous cellular phone reviews dataset was extremely favorable.	The Naive Bayes method is used to overcome the problems of declension suffixes, different word spacing, and conversational terms.
Faranak Ebrahimi Raashed et al. [172]	Create a sentiment dictionary; the proposed technique achieves 80% accuracy.	supervised sentiment classification approach for Persian sentiment classification
Hamoon Jafarian et al. [317]	In state-of-the-art investigations, the system improves the performance of ABSA with 91% accuracy, which is 5.5% higher than the Pars-ABSA dataset.	ParsBERT is a BERT for improving Aspect-Based Sentiment Analysis for the Persian language.
Kia Dashtipour et al. [303]	For the hotel dataset, the movie dataset, 2D-CNN, achieved an accuracy of 89.76%.	A stacked bidirectional-LSTM deep learning technique for Persian SA.
Kia Dashtipour et al. [306]	There are 3,000 reviews in the hotel review collection. The experiment reveals that the present method achieves 79.68% accuracy.	Preprocessing is accomplished by normalization, tokenization, and stemming, with shallow and deep learning groups of classifiers used to improve the state-of-the-art Persian SA method.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mohammad Heydari et al. [315]	A dataset of 100,000 customer evaluations from several categories yielded a 78.3 F1 measure score on different polarity.	Preprocessing included normalization and tokenization, as well as a hybrid technique that used deep learning for Persian SA.
Kia Dashtipour et al. [349]	The results suggest that the present vocabulary, as opposed to the PerSent Lexicon, is beneficial for sentiment analysis.	Increase the lexicon of PerSent Persian SA by more than 1,000 colloquial terms with their polarity.
Kia Dashtipour et al. [170]	The best performance was obtained by integrating unigram, bigram, and trigram features from the Persian Movie Review dataset with the PerSent Lexicon.	SVM classifier for Persian SA using N-gram and part-of-speech tagging.
Rouzbeh Ghasemi et al. [351]	Using Amazon and Persian Digikala datasets, the outcome showed a 22% increase in Persian SA.	Cross-lingual deep learning framework that makes use of the English language's available training data.
Batoul Botshekanan Dehkordi et al. [195]	The bipolar word, Digikala corpus, can improve the lexicon-based strategy for detecting two polarities and anticipating score difficulties.	method for determining polarity in phrases with one positive and one negative word
Rahim Dehkharghani [229]	The outcome demonstrates that the existing SentiFars Lexicon for Persian is efficient and beneficial.	The translation-based approach to the Persian polarity lexicon.
Mohammad Ehsan Basiri et al. [192]	The dataset PerView, According to the results, PerLex outperforms the current CNRC, Adjectives, SentiStrength, PerSent, and LexiPers lexicons. Reduce the existing approach's execution time as well.	Improve Persian SA by refining the vocabulary and utilizing PerLex terms to train machine learning algorithms.
Milad Vazan et al. [344]	The Persian Review dataset results demonstrate that CNN and GRU models are given more priority than LSTM and Bi-LSTM models.	Aspect-Category-Detection (ACD) and Aspect-Category-Polarity (ACP) difficulties have been solved.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Kia Dashtipour et al. [304]	In the Persian multimodel dataset, the result reveals contextual merging of multimodal information such as textual, audio, and visual features to achieve 91.39% accuracy, which is greater than the unmodel's 89.24% accuracy.	Context aware Multimodal S A, for Persian SA.
Nazanian Sabri et al. [331]	BERT, Yandex Dictionary, pretrained word embedding results reveal a 66.17% F1 score of 63.66 on the data.	For Persian SA, employed a code-mixed dataset and a neural classification model.
Seyed Arad Ashrafi Asli et al. [257]	The result indicates remarkable performance compared to previous methodologies. The F1 score was 80 percent accurate using of LSTM and CNN.	MIrasOpinion, a large Persian SA dataset, using an active learning method known as LDA sampling for text-based categorization.
Rahim Dehkharghani et al. [267]	The results suggest that the present system is more efficient than the previous technique.	Deep neural networks for Persian sentiment analysis to overcome challenges such as intensification, negation, and distinct granularity level.
javed PourMoustafa Roshan sharami et al. [287]	The author's goal was met by augmenting three Persian SA corpora.	Preprocessing included the following steps: normalization, punctuation removal, single character removal, digit removal, and lemmatization, as well as the use of LSTM and CNN for extracting opinion from individual phrases using the neural word embedding approach.
Behnam Sabeti et al. [248]	Display credible accuracy and F1 score in Persian SA.	LexiPers is a Persian SA lexicon built with k-nearest neighbor and a novel graph-based technique.
Kia Dashtipour et al. [263]	10–16% and 3–4%, respectively	Preprocessing is accomplished by normalization and tokenization, employing a hybrid technique based on SA and Dependency Grammar, deep learning, and integrated linguistic rules to improve polarity identification.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Taha Shangipour ataie et al. [250]	The outcome demonstrates the equivalent to the English state-of-the-Art technique.	Pars-ABSA aspect dataset for Persian SA.
Kia Dashtipour et al. [196]	Persian movie review dataset, the result demonstrates that the present system outperforms state-of-the-art MLP.	Deep autoencoders and deep convolutional neural networks
Pedram Hossenini et al. [200]	SentiPers is the name of the corpus.	The produced dataset corpus has almost 26,000 sentences.
Mohammad Bagher Dastgheib et al. [264]	Sentiment categorization result for two domains more than 10%	Convolutional Neural Network (CNN) with Structural Correspondence Learning (SCL) for Persian SA.
Ayoub Bagheri et al. [107]	The present technique is successful and efficient, according to the cell phone reviews dataset.	Preprocessing done tokenization, lemmatization, and feature selection. Many colloquial words are used to overcome declension suffixes and word spacing difficulties.
Zahra Bokaee et al. [246]	The system achieved an accuracy of 85% on both models using two Persian Twitter and Digikala datasets.	CNN-LSTM deep learning hybrid for Persian SA.
Mohammad Ehsan Basiri et al. [108]	demonstrates that the system outperforms state-of-the-art supervised machine learning methods	In Persian polarity lexicon, and analyzed the system's effectiveness for classifying the polarity using online cell phone reviews.

2.2.8 Named Entity Recognition

Saeede Momtazi et al. [278] propose a system for Persian name entity recognition founded on a deep neural network for entity tagging and word representation. Word2Vec and the skip-gram model were used to determine word similarity. Fast text is used to compare words for similarities depending on word context. BiLSTM and LSTM were used to predict entities from text history to detect entities. The system is used with a brand-new dataset that has 15 different labels for person groupings, including person, organization, faiths, locations, books, magazines, languages, movies, nationalities, jobs, events, dates, fields, and others. The system obtained 72.92 F-score.

Atefeh Zafarian et al. [141] shifted the feature to a low-resource language using a rich resource, unlabeled corpus. The proposed technique employs graph-based semi-supervised learning to train a CRF-based supervised classifier using labeled data while foretelling with high confidence using unlabeled data. Lastly, a new training set improved the Named Entity Recognition (NER) model's performance.

Kia Dashtipour et al. [171] bring in a new robustness. The Probabilistic Named Entity Recognition (PNER) framework can recognize name entities such as location, name, and date in Persian. The suggested strategy makes use of machine learning, language rules, and a unified named entity dictionary. The machine concludes by evaluating the manually annotated movie review. The outcome demonstrates that the existing system outperforms a cutting-edge rule-based method. Future writers have a propensity to use more tags and systems that can recognize name entities across several languages.

Leila Hafezi et al. [199] describe a method for neural network-based named entity recognition in Persian. Pre-trained word embeddings and the BILSTM-CRF model were employed in the suggested technique. In the current technique, the author employed two datasets, the first of which contains 709 news stories with 302,530 tokens, and, 41,148 name entities labeled tokens. The second has 250,015 tokens and 7.683 Persian text corpora created using the CoNLL and MUC standard datasets. The system ultimately acquired an F1 score of 86.55%, which is higher than the performance in Persian NER. The present method produces an F1 score of 84.23 in the second dataset trial, which is 5.78% higher than Poostchi's prior method's output.

Hanieh Poostchi et al. [205] a method for deep learning-based entity recognition of Persian names is presented. The ArmanPersoNERCorpus[37], the first Persian named entity recognition corpus, was employed in the suggested technique. The author employed four different word embeddings in the present system, based on CBOW, GloVe, skip-gram, and HPCA. The recommended strategy received an F1 score of 77.45 percent; the present F1 measure is higher by 12.32 than the prior best outcome strategies.

Mohammad Ebrahim Khademi et al. [274] provide a methodology for unsupervised Persian sentiment analysis. The suggested solution took advantage of name entity detection and summary generation. Uses the Arman name entity database to train neural networks and the Hamshahri2 corpus for word embedding. The recommended methodology was then tested on the Pasokh single-document corpus using ROUGE. The method improved the ROGUE-2 recall score by 10.2 percent.

Shadi Hosseninnejad et al. [174] introduce the A'laam corpus, a standard name entity corpus for the Persian language. 13 NE tags have been applied to almost 250,000 tokens in the present corpus. The corpus was split into test and train parts for evaluation, with 90% of the corpus being the train section and 10% being connected to the test section. The results demonstrate 92.94-percent precision and 78.48 percent recall utilizing the conditional Random Field and Persian NER system.

Majid Asgari-Bidhendi et al. [299] familiarize the ParsNER-Social corpus for named entity recognition training, which is derived from social media and ten telegram channels divided into ten categories. The current corpus

[37] https://github.com/AminMozhgani/Persian_NER

includes 205,373 tokens, and their NER tags. Corpus from the ParsBERT+BERT-TokenClass model achieved an F1-score of 89.65% respectively.

L. Jafar Tafreshi et al. [271] presented an approach for Persian named entity recognition based on conditional random fields for building models. The system used a syntactic dependency treebank with almost 30,000 sentences, which was created in Islamic sciences research center. The result shows the system can achieve 86.86% precision, 80.29% recall and overall 83.44% F1 score. Which is higher than reported previously in the domain of NER.

Mojtaba Zali et al. [216] offer a neural network-based method for recognizing Persian-named entities. The suggested solution made use of word vector representation, which not only helped to demonstrate how a word was embedded with features, but also helped to address the issue of feature selection. Alongside, offered guidance on how to create an efficient system with the least amount of domain-relatedness.

Mohammad Hadi Bokaei et al. [193] deep learning-based improvements to the identification of Persian-named entities. ArmanPersoNER, which has 7,682 phrases and 250,015 tokens, was utilized by the author to extract the feature and train model. RNN and CNN models are applied in the suggested feature extraction technique. Find the highest sequence tag for word sequence input after applying a conditional random field. The experiment's findings indicate an improvement over the corpus's present F1 score of 81.50 percent on the word level and 76.79 percent on the phrase level. The best model demonstrates that more review of the existing corpus is required to increase performance, which also experienced errors in the output.

F. Balouchzahi et al. [259] offer a Transfer Learning (TL) model-based Universal Language Model Fine-tuning (ULMFiT) for Persian-named entity recognition. The suggested approach, called PUNER, is trained using data from Wikipedia that was collected in the general domain. The manPersoNERCorpus and PersionNER datasets were used to test the existing methodology. The system scored 92.82 F1 on the ArmanPersoNERCorpus and 82.16 F1 score on the Persian-NER corpus, according to the results. The author compares and contrasts the ML learning strategy with the existing system using BiLSTM and five-word embedding models. The comparison revealed that TL performs as well as DL models and better than ML. Other languages outside Persian can use ULMFiT Persian LM weight.

Hanieh Poostchi et al. [157] provide the manually annotated ArmanPersoNERCorpus for Persian-named entity recognition. The NER pipeline is also introduced for the Persian language in the suggested technique with corpus. Word embedding by Hellinger PCA and classification by structural SVM-HMM classifiers make up the pipeline's major part. The present corpus received an F1 score of 72.59 in MUC7 and 67.13 in CoNLL during the trial. The present outcome is superior to CRF and Jordan-RNN.

Farid Ahmadi et al. [125] Describe a machine learning and rule-based method for recognizing Persian Name Entities. Machine learning techniques Viterbi and HMM algorithms were employed in the suggested strategy. For the rule-based method of NER identification, a collection of lexical resources and a pattern-based were also utilized. The experiment system used a corpus of 32,606 tokens to obtain 89.73-percent accuracy, 82.44-percent recall, and an overall F1 score of 85.93 percent.

Mohammad Mahdi Aghajani et al. [296] create a corpus from Twitter that is gathered for ParsTwiNER[38]. The current corpus, which was produced based on the MUC-6 or CoNLL 2004 standard, has 250K tokens. Cohen's Kappa coefficient displays a high score of 0.95 for annotator consistency. As well as using BERT architecture, they trained new models using it. The outcome shows that the approach is effective in both formal and casual Persian.

Frame Jalali Farahani et al. [311] By combining transfer learning and active learning strategies, they introduce BERT-PersNER for Persian-named entity recognition. For the decoding tag, the author employed a condition Random Field, however, the performance of BERT-PersNER was exceptional. The author utilized the Peyma and Arman databases[39] in the experiment. Using the present supervised learning strategy, the system attained an accuracy of 92.15 percent and proficiency of 92.41 percent.

Mahdi Mohseni et al. [242] introduce MorphoBERT, a method for recognizing Persian names using morphological analysis and BERT. BiLSTM was utilized to label the tokens on the vector representation. The system's recall and F1 measures of 87.0, 83.8, and 8.4 respectively are the highest of all contributor systems.

Nasirn Taghizadeh et al. [288] reported the NSURl-2019 take 7 for the identification of Persian Named Entities. With an accuracy rate of 85.4% at the phrase-based level, the MorphoBERT system demonstrated the highest proficiency of competency. The BERT model and Bi-LSTM were used by the system to account for morphological aspects of Persian words.

Ehsan Taher et al. [289] provide a method for recognizing entities by their Persian names that is based on a pre-trained deep bidirectional network. In the NSURL-2019 Take 7 competition, the system was examined, and it came in second. In the examination of words and phrases, the F1 measure had a CoNLL score of 83.5 and 88.4, respectively.

Mahsa Sadat Shahshahani et al. [211] introduce the PEYMA standard corpus for the identification of named entities in Persian. The data used in the present corpus is scraped from news websites. There are 302,530 tokens in 709 documents from the corpus. The 41,148 tokens make up the labeled entity. An accurate F1 score of 84 percent was obtained for the corpus evaluation utilizing a hybrid technique that included rule-based and statistical components. Place, person, date, organization, percentage, currency, and time.

Morteza Kolali Khormuji et al. [117] offer a method for Persian named entity recognition based on local filters and various dictionaries. Using Approximate String Evaluation Metric (ASEM), the experiment and evaluation system was able to obtain up to 88.95-percent precision, 79.65-percent recall, and an overall F1 score of 82.73 percent. [40] Mohammad Mahdi Abdollah Pour et al. [347] using a number of transformer-based language mod-

[38] https://github.com/overfit-ir/parstwiner

[39] https://github.com/kargaranamir/Persian-Datasets

[40] Approximate String Evaluation Metric (ASEM) Persian is a head-initial language. The head is in the right position, since the Persian transliteration reads right to left. Therefore, it is important to think about a nominal group's right border. Here, choosing the appropriate border type is desired [272]

1. Name سعدالله کریمی

els that the author has altered with different classifiers, they evaluate these models on the Persian NER task. they carry out a comparative study to assess the impact of text representation and text categorization methods on Persian NER performance. To train and evaluate the models, they employ three distinct Persian NER datasets: MoNa, Peyma, and Arman. Experimental results showed that the best performance was achieved by XLM-R with a Conditional Random Field (CRF) layer and a linear layer. This model produced phrase-based F-measures of 70.04, 86.37, and 79.25 on the MoNa, Peyma, and Arman datasets, as well as word-based F1 scores of 78, 84.02, and 89.73. These results demonstrate cutting-edge Persian NER task performance [354].

Table 2.12: Persian Named Entity Recognition

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Saeede Momtazi et al. [278]	The system achieves 72.92 F1 measure from a dataset of 15 different labels for Persian NER.	Deep learning approaches like as Word2Vec and skip-gram, BiLSTM, and LSTM are used to identify Persian NER.
Atefeh Zafarian et al. [141]	NER's performance improves as a fresh training set is created.	Shifting from a rich resource language to a low resource language, a Graph-based supervised classifier with high confidence foretelling was employed to construct a new Persian NER training set.
Kia Dashtipour et al. [171]	The system detected NER in multiple languages using a movie review dataset.	PNER framework is a hybrid technique that uses a unified NE dictionary, machine learning, and grammatical rules to develop a powerful NER for Persian.
Leila Hafezi et al. [199]	Two Persian NER corpora, the second dataset yields an 86.55% F1 score, whereas the present technique yields an 84.23 F1 score.	For Persian NER, used the BILSTM-CRF model and pre-trained word embedding based on neural network architecture.
Hanieh Poostchi et al. [205]	ArmanPersoNERCorpus, outcome demos 77.45% F1 score.	CBOW, GloVe, skip-gram, and HPCA word embeddings for Persian NER based on deep learning architecture.

[2] Organization سازمان جهانی غذا امنیت غذایی و رفع گرسنگی است

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mohammad Ebrahim Khademi et al. [274]	The system improved its ROGUE-2 recall score by 10.2% using the Hamshahri2 corpora, Arman Name entity corpus, and Pasokh single document corpus.	Neural network for Persian NER, unsupervised technique for Persian Named Entity Recognition.
Shadi Hosseninnejad et al. [174]	The results suggest that A'laam Corpus has 92.94% accuracy and 78.48% recall.	Using conditional random fields and the Persian NER system, create a standard corpus for Persian NER.
Majid Asgari-Bidhendi et al. [299]	The ParsBERT+BERT-TokenClass model corpus earned an F1-score of 89.65%.	For Persian NER, created the ParsNER-Social corpus.
L. Jafar Tafreshi et al. [271]	The results reveal 86.86% precision, 80.29% recall, and a total F1 score of 83.44.	For Persian NER, conditional random fields and a syntactic dependency treebank were used.
Mojtaba Zali et al. [216]	The system is used to create an effective system that is less relevant to the domain.	Utilized Neural network, vector representation of word for Persian NER
Mohammad Hadi Bokaei et al. [193]	The ArmanPersoNER corpus results show improvement in the F1 score of 81.50% and 76.79% on the word and phrase levels, respectively.	RNN and CNN models were used after the conditional random field to find the top sequence tag for the sequence input using deep learning for Persian NER.
F. Balouchzahi et al. [259]	manPersoNERCorpus, Persian NER for evaluation, the system obtained 92.82 F1 scores for ArmanPersoNER Corpus, 82.16 F1 scores for Persian NER Corpus, and the TL method is better than ML and equal performance with DL model. ULMFiT Persian LM weight may be utilized in other languages aside from Persian.	ULMFiT for Persian NER, commonly known as PUNER-Persian, was built using Transfer Learning TL.
Hanieh Poostchi et al. [157]	The corpus obtained 72.59 in MUC7 and 67.13 in CoNLL, which is higher than CRF and Jordan-RNN.	For Persian NER, the Structural SVM-HMM classifier was used for corpus experimentation, and word embedding using Hellinger PCA was used to create the ArmanPersoNER corpus.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Farid Ahmadi et al. [125]	The result reveals 89.73% precision, 82.44% recall, and an overall F1 score of 85.93% while using a corpus of 32,606 tokens.	The Viterbi and HMM algorithms, which are machine learning approaches, used lexical resources and patterns based on the identification NER, which is a rule-based approach for Persian NER.
Mohammad Mahdi Aghajani et al. [296]	The study revealed that the model performs well in both formal and informal Persian, with Cohen's Coefficient showing a high score of 0.95 in annotation consistency.	ParsTwiNER corpus was constructed based on the MUC-6 or CoNLL 2004 standards, and new models were trained using the BERT architecture.
Frame Jalali Farahani et al. [311]	Using the Peyma and Arman datasets, the results demonstrate 92.15 and 92.41 percent proficiency, respectively, based on the present supervised learning technique.	For the decoding tag, transfer learning and active learning methods were used, as well as a conditional Random Field. BERT-PersNER was created for Persian NER.
Mahdi Mohseni et al. [242]	Systems achieve recall and F1 measures of 87.0, 83.8, and 8.4, respectively, which is higher than any other contributing system.	For the current MorphoBERT system, Morphological Analysis and BERT, BiLSTM were used to tag tokens on vector representation.
Nasirn Taghizadeh et al. [288]	The MorphoBERT system achieves the highest proficiency of 85.4% accuracy at the phrase-based level.	Report NSURI-2019 task 7 for Persian NER using BERT and Bi-LSTM for Persian morphological NER.
Ehsan Taher et al. [289]	The system came in second place in NSURL-2019 task 7 with F1 measures of 83.5 and 88.4 CoNLL score in word and phrase level evaluation, respectively.	For Persian NER, a pre-trained deep bidirectional network was used.
Mahsa Sadat Shahshahani et al. [211]	The present method achieved 84% accuracy in the F1 score when evaluated utilizing a hybrid approach of rule-based and statistical parts.	PEYMA, a dataset for Persian named entity recognition, was created.
Morteza Kolali Khormuji et al. [117]	Using ASEM, the system achieved 88.95% precision, 79.65% recall, and a total F1 score of 82.73%.	For Persian NER, various dictionaries based on local filters were employed.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mohammad Mahdi Abdollah Pour et al. [347]	Three distinct Persian NER datasets, MoNa, Peyma, and Arman This model achieved phrase-based F-measures of 70.04, 86.37, and 79.25 on the MoNa, Peyma, and Arman datasets, and word-based F scores of 78, 84.02, and 89.73. These findings indicate state-of-the-art Persian NER task performance.	A comparison of text representation and text categorization approaches to examine their influence on Persian NER performance

2.2.9 Text Summarization

Martin Hassel et al. [14] provided a method for summarizing Persian text called FarsiSum. The system is a client/server application that was created using the Perl programming language. Currently, there are three processes involved in summarizing a text or document. To start, tokenize the text. In the second stage, words are compared with the stop-list words and then scored before being converted from ASCII to UTF-8. The summary eventually reaches the HTTP server, which then displays the content to the client via the user interface.

Hassan Shakeri et al. [86] provide a graph-based method for summarizing Persian texts. Using a text or corpus, the suggested graph-based algorithm is utilized to extract significant sentences. The outcome demonstrates that the summary phrases are connected to one another and that it is close to the human summary version. Evaluation reveals that the present approach improves ROUGH-1, precision, and recall better than FarsiSum.

Marzieh Berenjkoub et al. [69] provide a supervised machine learning-based method for summarizing Persian text. The proposed approach is designed to condense the scientific article. Texts are broken up into structural divisions by authors, such as an abstract, an introduction, a method, an experiment, and content from the aforementioned sections. The system's assessment demonstrates that integrating surface, relevance, and content features all at once was beneficial for the method part and the full summary, with the introduction and experimental portion producing the best results when employing content features.

Farshad Kiyoumarsi et al. [50] Persian text summarization has been enhanced using a fuzzy logic method. The suggested procedure was utilized to survey cutting-edge techniques to summarize Persian literature. Moreover, tried to overcome the deficiency of one machine learning approach for making an automatic text summarizer for summarizing text based on fuzzy logic to obtain a good result, differing from old method. The present approach outperformed the old method in Persian text summarization by an average of 85%.

Mehrnoush Shamsfard et al. [38] introduce PARSUMIST, which uses lexical graphs and chains, statistical techniques, and heuristic techniques to extract meaningful phrases from inputs to summarize Persian text. The system was effective, and the output summary is comparable to human summarizing, according to the performance evaluation.

Marzieh Berenjkoob et al. [30] propose a method for Persian text summarizing based on Persian text stemming. Unlike other methods, the suggested solution does not have a constraint for dealing with irregular plural word issues. A Dehkhoda dictionary, one of the most well-known Persian dictionaries, was used to evaluate the existing system. It has a recall number of 3 and an accuracy of 70% when summarizing Persian texts.

Saeed Farzi et al. [231] Described a method for summarizing Persian text that relies on a semi-supervised technique. The algorithm uses the Bistoon corpus, which has 14,000 human-made summaries of items. It is a Persian news summarizer under the name of Katibeh. Ijaz, a system tested with a cutting-edge news summarizer, was used to analyze two typical summarizing corpora. The results demonstrate that the present system, which has more accuracy, ROUGE metrics, F-measure, and a human decision-maker, is superior to Ijaz.

Asifpoor Masoumi et al. [120] introduce the Ijaz method, which is a method for summarizing Persian texts. The suggested technique for mechanically summarizing Persian texts in a single document is experiences from comparable internal and external systems have been applied to this system. The present method uses a sizable standard corpus for the first time to summarize Persian news. The precision and recall of the system have greatly increased.

Tayyebeh Hosseinikhah et al. [201] provide a graph similarity-based NLP method for summarizing Persian text. The developers of the proposed method use text mining and NLP to improve the accuracy of the summaries. In the present method, part-of-speech tagging is also utilized. Approaches can be used to extract significant and meaningful words and phrases that improve the accuracy of the system. A similarity strategy is utilized to reduce the redundancy issue and choose the sentence graph.

Azadeh Zamanifar et al. [65] introduce the AZOM method, which summarizes Persian texts. The statistical and semantic properties of text were utilized in the suggested technique. The method is also applicable to unstructured documents that have been summarized. The technique is used with Persian, but it may be used with any other language as well. The experiment concludes by demonstrating that the system works better than the existing standard text summarizer for Persian.

Mohsen Tofighy et al. [64] provide a fractal-based method for summarizing Persian texts. The present strategy is to utilize documents in hierarchical manner. The outcome of the trial demonstrates that the approach is useful for improving extractive summarization proficiency.

Sayede Azadeh Hosseinzadeh et al. [114] provide a neural network-based method for summarizing Persian text. For training the current technique, 100 documents from the Hamshahri news corpus are employed. A test using the present dataset reveals a document summarization F1 score of 0.67.

Fatemeh Hojati Kermani et al. [233] propose a statistical, semantic, and heuristic-based strategy for summarizing Persian text. The suggested method first preprocesses the input before extracting feature vectors from significant phrases using the aforementioned techniques. The essay or article title, pronouns, proper nouns, sentence length, the topic word, keyword placement in the sentence, questions, and English words all play a significant role in the main feature selection scoring. The experiment demonstrates that its technique outperforms existing Persian summarizers in terms of ROUGE-N, recall, and F-measure.

Asef poormasoomi et al. [57] provide a context-based summary of many documents in Persian. To reduce redundancy and determine how similar ranking phrases are to one another semantically, the approach employed WordNet and LSA. Using the existing method can significantly improve accuracy, the trial concludes.

Hossein Shahverdian et al. [185] suggested a technique that combined graph-based summarization with lexical and semantic similarity. Comment uses clustering to manage many topics in a dataset. The experiment system displays an average score of 8.75 out of 10, raising the modern summarizer score to roughly 14 percent.

Elham Mahdipour et al. [119] recommended using a graph-based, TF-IDF method to weigh the sentences after word stemming. Simulated Annealing (SA) is used to choose the best sentences to summarize using the Genetic Algorithm (GA). The readability, topic-relation, and coherence factors serve as the foundation for the fitness function. The experiment also demonstrates the effectiveness of the existing strategy.

Hossein Kamyar et al. [49] when summarizing text, statistical aspects of text elements such as the frequency of words receives the most focus. To identify semantic chains in a text, the author of this study applied the centering theory. For introducing a brand-new automated approach to summarizing one document. The centering theory is utilized to constrict the processing pipeline and produce a cogent summary. The pipeline for current systems has multiple constituent lines for part-of-speech tagging, co-reference resolution, and semantic role labeling. The author used the Hamshahri2 corpus, which contains about 400 documents in TDT TREC format that were collected from the Hamshahri newspaper between 1997 and 2007. In comparison to FarsiSum, an LSA-based, Random-based summarizer with a high F1-measure from ROUGE1, ROUGE2, and ROUGE-SU4, the current technique gets more than 0.4 out of 0.5.

Azadeh Zamanifar et al. [29] The approach avoids comparable phrases based on word synonymy and instead looks at the text's conceptual properties. To discover compound words while maintaining the unity of the summary text, the author employed a dataset that contained more than 1,400 Farsi combined terms. The hybrid system outperforms FarsiSum, a well-known Farsi summarizer, by combining statistical aspects of documents with Farsi linguistic features.

Mohammad Ebrahim Khademi et al. [275] provide a continuous vector space-based method for summarizing Persian texts. The suggested technique creates a word frequency dictionary by first generating word embedding using the Hamshahri2 dataset. Extract the document's keyword from the corpus after using the current approach. Sort the sentences into categories to create the summary. The author employed Pasokh, a one-document corpus, for the experiment using the present methodology and used the ROUGE assessment measure. The outcome displays superior performance in relation to ongoing work as compared to the state-of-the-art. Following that, the system received an overall score of 7.5% ROUGE-2 recall as compared to the supervised Persian method.

Seyyed Mohsen Tofighy et al. [105] provide a method for summarizing Persian material that is based on the Analytic Hierarchy Process (AHP). The suggested strategy uses a weighting and combination approach to improve the quality of the summary of Persian language material. The present method outperforms other Persian text summarization techniques already in use.

Alireza Salemi et al. [335] provide a method for abstractive Persian text synthesis. The author suggested ARMAN, an encoder-decoder model based on transformers. Based on updated semantic scores, the ARMAN is used to identify critical sentences for papers, which are disguised as a pseudo-summary. The author employed modified sentence reordering in the summary version of this approach, which is comparable to human pattern writing. System tested by the author using six earlier techniques for summarizing Persian texts. Results indicate that all six downstream procedures, which are measured by ROUGE and BERTScore, received state-of-the-art proficiency utilizing current approaches. The approach also performs better than multiple-choice, response, textual entailment, and question paraphrasing. The human evaluation test technique demonstrates that applying semantic ratings may fully improve summary outcomes.

Fatemeh Shafiee et al. [212] provide a summary in Persian of a single document and many documents. The suggested approach goes after feature extraction. Persian WordNet divides the input into three unique clusters, relatedness, similarity, and coherency, to produce a qualitative summary by utilizing FarsNet to uncover semantic information about the word. The FarsiSum, PARSUMIST, and Ijaz approach test. Precision, recall, F1-measure, ROUGE-N, and ROUGE-L all showed that the present method was superior to these three approaches [354].

Table 2.13: Persian Text Summarization

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Martin Hassel et al. [14]	A summary is displayed to the client via the user interface.	Tokenization is completed first, followed by keyword extraction using scoring, and conversion of the ASCII format to UTF-8. The system is a work client/server application built in Perl for Persian text summarizing.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Hassan Shakeri et al. [86]	The results reveal that the summary phrases are related to each other and that the outcome is close to the human summary version. The assessment shows that the system improves accuracy, recall, and ROUGH-1 better than FarsiSum.	A graph-based technique was used to extract key sentences from a text or corpus summarization of a Persian Text.
Marzieh Berenjkoub et al. [69]	The results reveal that employing surface, relevance, and content characteristics combined was beneficial for the technical part and the full summary, with content features for the introduction and experimental sections producing the best results, using scientific articles.	For Persian text summarization, a supervised machine learning technique was utilized.
Farshad Kiyoumarsi et al. [50]	When compared to the prior summarizing approach for Persian, the present method achieved an 85% improvement in accuracy.	The fuzzy logic technique was used to improve the quality of Persian text summarization.
Mehrnoush Shamsfard et al. [38]	The end product is a solid overview. The summary approaches human summarization.	For selecting suitable phrases for input, lexical graphs and chains were employed, as well as a statistical technique combined with a heuristic strategy. Create PARSUMIST for summarizing Persian text.
Marzieh Berenjkoob et al. [30]	In summarizing Persian literature, employed the Dehkhoda lexicon and got an accuracy of 70% and a recall of 3.	Stemming was employed for Persian text summarizing.
Saeed Farzi et al. [231]	Using the Bistoon corpus, which comprises 14,000 human-created summary materials, the present system outperforms Ijaz in terms of accuracy, ROUGE metrics, F1-measure, and human decision-maker.	For testing, they employed a semi-supervised technique using Ijaz's cutting-edge news summarizer system. In order to offer Katibeh Persian text summation.
Asifpoor Masoumi et al. [120]	Standard corpus for summarizing Persian news. The results suggest that accuracy and recall improve significantly.	Ijaz, a Persian automatic text summary for a single document text, is presented.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Tayyebeh Hosseinikhah et al. [201]	Methods aid in the extraction of significant and relevant words and phrases, allowing the system to improve accuracy and reduce redundancy.	For Persian Sentiment analysis, graph resemblance, NLP and text mining methods, and part-of-speech tagging were used to extract relevant and meaningful words.
Azadeh Zamanifar et al. [65]	The system outperforms previous Persian text summarizers.	AZOM is a Persian text summary tool that uses statistical and semantic text properties to summarize unstructured materials.
Mohsen Tofighy et al. [64]	The strategy proved effective for improving extractive summarization skills.	For Persian text summary, fractal theory was used, which operates in a hierarchical framework.
Sayede Azadeh Hosseinzadeh et al. [114]	The Hamshahri news corpus was used, and the result indicates a 0.67 F1 score on document summarization.	For Persian text summarization, neural networks were used.
Fatemeh Hojati Kermani et al. [233]	The ROUGE-N, recall, and F1 scores are better than previous Persian summarizations.	For Persian Text summarization, first do preprocessing and then extract feature vectors on important sentences using statistical, semantic, and heuristic approaches.
Asef poormasoomi et al. [57]	The results suggest that applying the present technique significantly improves accuracy.	For Persian multi-document text summarization, utilized WordNet and LSA to reduce redundancy.
Hossein Shahverdian et al. [185]	For summarizing, employ a large corpus of user-generated comments in Persian. The outcome is an average score of 8.75 out of 10, raising the state-of-the-art summarizer score to over 14%.	For the Persian language, semantic, lexical similarities, and graph-based summarization were used.
Elham Mahdipour et al. [119]	For Persian text summary, the fitness function is based on readability, topic-relationship, and cohesion factors. The outcome demonstrates that the existing technique is effective.	Using a graph-based, TF-IDF technique, the phrases were first stemmed using a Genetic Algorithm (GA) and Simulated Annealing (SA)

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Hossein Kamyar et al. [49]	Using the Hamshahri2 corpus, the F1 score is greater than 4 out of 5 when compared to FarsiSum and LSA-based summarizers, as well as random-based summarizers in ROUGE1, ROUGE2, and ROUGE-SU4.	Pipeline processing and extraction of a coherent summary, the centering theory, detects semantic chins in a text using statistical features. The centering theory was applied for Persian text summarizing.
Azadeh Zamanifar et al. [29]	For recognizing compound words, a dataset of 1400 Farsi compound words was used. Outperform FarsiSum, a well-known Farsi summarizer, by using statistical aspects of documents coupled with Farsi linguistic features.	Preprocessing: The System analyses the conceptual property of the text, eliminating comparable sentences based on word synonymy, by applying the Farsi stemming algorithm and deleting stop words.
Mohammad Ebrahim Khademi et al. [275]	Using the Hamshahri2 and Pasokh corpora, the results demonstrate that in contrast to supervised approaches, the total accuracy of ROUGE-2 recall is 7.5%. Better than the existing approach's state-of-the-art.	Used continuous vector space for Persian text summarization.
Seyyed Mohsen Tofighy et al. [105]	When compared to the existing Persian summarizer, the result demonstrates good performance.	To improve the quality of Persian summarization, used the Analytic Hierarchy Process (AHP) using a weighting and combination technique.
Alireza Salemi et al. [335]	Six downstream datasets were used, including PN-Summary, Wiki-summary, VOA Dataset, PerKey, Tebyan, and ParsiNLU Datasets. Describing that utilizing semantic ratings improves summarization outcomes by establishing human evaluation.	For Persian text summarization, ARMAN, a Transformer-based encoder decoder model, was used.
Fatemeh Shafiee et al. [212]	The present method outperforms FarsiSum, PARSUMIST, and Ijaz in accuracy, recall, F1 score, ROUGE-N, and ROUGE-L.	FarsNet may be used to find semantic information about a term in a Persian single or multi-document summary.
Mehrdad Farahani et al. [313]	Introduce the PN-Summary dataset in the evaluation of the present dataset; the outcome was promising due to the lack of previous work in this area.	For Persian text summarization, multilingual T5 and the ParsBERT model were used.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Hosein Rezaei et al. [247]	In the second trial, using Pasokh Corpus, the results indicate 0.78, 0.71, and 0.74 in summary quality.	For supervised single document summarization in Persian, an extractive approach with vector representation was used.
Saeid Parvandeh et al. [155]	The results reveal that tiny centrality measures beat three strong unsupervised baselines when evaluated ad hoc using ROUGE scores on current summaries.	The author describes three distinct unsupervised approaches for multi-document and single-document summarizing in Persian, including PARSUMIST, sentences networks, and centrality measures.
Behdad Behmadi Moghaddas et al. [100]	Pasokh offers numerous summaries as examples for the experimentation of a one-document and multi-document summarizing system.	Pasokh, a standard corpus for Persian text summarization, was created.
Mehrnoosh Bazrfkan et al. [109]	Each sentence's value represents how vital it is to include that sentence in the final summary. The three membership functions on which the value is based are L, VL (low and very low), M (average), H (high), and VH (very high).	For Persian text summarization, they used a machine learning approach that enters data into a fuzzy system.
Mehdi Bazghandi et al. [68]	Using a corpus of Persian sport news items, the algorithm achieves greater accuracy than other approaches.	For optimizing the approaches, desirable clustering, and collective intelligence algorithms are applied.

Mehrdad Farahani et al. [313] show two pre-trained Persian text summarizations, one of which is a BERT2BERT warm-started from the ParsBERT language model and the other of which uses the Multilingual T5. To handle Persian text summarizing using an abstract method. Since there don't appear to be any datasets for text summary in Persian, the authors present PN-Summary, a brand-new dataset for text summarization. The results of the suggested strategy utilizing the available dataset are better and more encouraging. Because there aren't many works in this discipline, the author of the few works done in it cannot be compared to any earlier work.

Hosein Rezaei et al. [247] provide a strategy for extractive-based Persian text summarization. The author employed the Pasokh corpus, which has roughly 2,500 sentences overall and 100 news Persian documents. The present approach unifies document characteristics into vectors of each unique phrase. This simple system can comprehend the context. To improve the accuracy. The model can produce a general and succinct summary for learning. In experiment two's summary quality, ROUGE1, ROUGE2, and ROUGE-L results reveal F1 scores of 0.78, 0.71, and 0.74.

Saeid Parvandeh et al. [155] To see how it performs on actual corpora that comprise news stories, the first author upgraded two of the most well-known and often quoted algorithms for summarizing Persian text. The author discusses two approaches based on phrase networks and centrality measures, as well as three distinct unsupervised systems for multi-document and single-document summary, the first of which was utilized in the renowned Persian text summarizer PARSUMIST. A human description of the experiment reveals that the graph-based technique performs better than the upgrading system. The author contrasts graph-based with graph-based in-depth by employing centrality metrics to provide several summaries. The outcome demonstrates that a small class of centrality metrics outperforms three powerful unsupervised baselines when utilizing ad-hoc assessment based on the ROUGE score on currently suggested summaries.

Behdad Behmadi Moghaddas et al. [100] present the Pasokh corpus for summarizing Persian texts. The present corpus was developed as a standard corpus for text summarization over the course of 2,000 man-hours of labor. For the experiment of single-document and multiple-document summarizing systems, the corpus contains a significant number of summaries as references.

Mehrnoosh Bazrfkan et al. [109] provide a machine learning-based method for summarizing Persian text. This method makes use of indices like sentence length, connected topics, and comparable keywords as the input to fuzzy systems. Thereafter, a knowledge base is used to summarize the whole collection of rules. Following this, each sentence is given a value between 0 and 1 based on the rules and features that are already present in the Knowledge Base. The importance of each statement in the final summary is shown by the value given to each phrase. The three membership functions L, VL (low and very low), M (average), H (high), and VH (very high) serve as the foundation for the value.

Mehdi Bazghandi et al. [68] provide a PSO Clustering-based extractive summarization method for summarizing Persian text. The semantic similarity of text is the foundation of the method world. After clustering, a phrase is selected from each cluster. A collection of Persian sports news stories were evaluated. The outcome demonstrates that certain methods are more accurate than others.

2.2.10 Spam and Fake News Detection

Neshat Safarian et al. [227] provide a method for Persian Spam Review identification based on Naive Bayes classifiers, support vector machines, and decision trees for spam classification. The suggested solution considers the accuracy of the existing system tested with specific feature composition and user evaluations of mobile phones obtained from the Iranian e-commerce website digikala.com. The highest accuracy of the decision tree classifier, as measured by the F1-measure of 0.778 in feature ranking, is shown in the result. In comparison to the other two classifiers, the decision tree performs best with an F1-measure of 0.824 when the positive feedback, total score, and feature polarity evaluation are combined.

Mohammad Mohsen sadr et al. [285] provided a method for detecting fake Persian Twitter news using machine learning. The suggested technique classified fake and real news using a corpus acquired from a Persian Twitter message utilizing K-NN, SVM, decision trees, random forests, and Multi-Layer Perceptrons (MLP). In the end, the learning algorithm's functional experiment and confusion matrix reveal the outcome. Accuracy scores of 90.25 and 90.20 were reached via randomized decision trees. 89.99% accuracy was attained with a random forest. The third level of precision was attained by the decision tree, which had an accuracy rate of 90.20 percent.

Masood Hamed Saghayan et al. [334] The author uses Persian Tweets regarding COVID-19 to reassure researchers about the efficacy of machine translation for false news identification. Following machine translation, on the aforementioned corpus, using the SVM classifier. 87 percent for binary class false news and 62 percent for Multiclass fake news identification. According to our statistics, machine translation has a negative influence on binary classification accuracy of 4% and multiclass classification of 23%.

Mohammad Ehsan Basiri et al. [228] Described a method for recognizing Persian spam using supervised machine learning techniques. The SpamPer corpus from digikala.com was utilized for the suggested technique. Decision tree classifiers, Naive Bayes, and support vector machines are used to classify data. The outcome demonstrates that the recommended strategy performs better when tested and trained on a balanced variant of the available dataset. Support vector machines are effective in classifying type one spam in experiments by utilizing metadata and constructed features, which produce encouraging results when used on unbalanced copies of the present dataset. Decision tree classifiers are superior to SVM and NB for type 2 spam.

Elahe Rabbani et al. [121] provide a method for identifying Persian web spam using content-based heuristics. Utilizing the Bag-of-Spam-Word technique for identifying web spam, the suggested system. To use the present method, the author developed a hosts Persian dataset. The BOSW technique and Support vector machine classifier had the best performance in identifying Persian website spam, according to an experiment.

Mohammadreza Smadi et al. [336] propose two methods for recognizing bogus news in Persian. The pre-trained BERT model is employed in the suggested strategy to identify bogus news. BERT-CNN is used to extract new features from embedding tokens that benefit from the BERT Model, while BERT-SLP is used to combine embedding of BERT for taxonomy. The TAJ dataset, a brand-new data set for fake recognition, is introduced by the author. They show that the BERT-SLP and BERT-CNN models perform better than historical benchmarks and conventional machine learning models. 15.58% and 17.1% higher accuracy in contrast to Zamani[187] reported result, and 11.29%, 11, 18% higher accuracy compared to Jahanbakhsh-Negadeh[352] reported result.

Mohammad Mohsen Sadr et al. [332] describe a method using an LSTM neural network for identifying bogus news on Persian Twitter. The proposed approach uses a 14-layer BLSTM neural network and LSTM to identify bogus news in Persian-language tweets. The final score is 92.05-percent performance ability, 91.10-percent recall, and 91.57 percent F1 score. The suggested system ultimately outperforms the above models in tests using

K-NN, random forest, Bayesian, linear regression, SVM, perceptron neural network, decision tree, AdaBoost probabilistic gradient, additionally tree, and gradient boost [354].

Table 2.14: Persian Spam and Fake Detection

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Neshat Safarian et al. [227]	A customer review of mobile phone results reveals that the decision tree classifier has the greatest accuracy. With an F1 score of 0.778 in ranking, the decision tree has the highest performance with an F1 score of 0.824 when compared to the other two classifiers.	For Persian spam review identification, used a decision tree, a Support Vector Machine, and naive Bayes classifiers.
Mohammad Mohsen sadr et al. [285]	Using the Twitter news corpus, the results demonstrate that randomized decision trees achieve 90.25 and 90.20 accuracy, whereas random forest achieves 89.99% accuracy while decision tree achieves 90.20% accuracy.	For determining whether the news is bogus or not, K-NN, SVM, decision tree, random forest, and MLP were utilized. For the identification of Persian forgeries.
Masood Hamed Saghayan et al. [334]	Persian tweets corpus about COVID-19, outcome reveals 62-percent for Multiclass fake new identification, and 87-percent for binary class fake news, machine translation has 4-percent negative impact on binary categorization accuracy and 23-percent negative impact on Multiclass classification, according to our data.	Measured whether machine translation is efficacious for fake news identification or not.
Mohammad Ehsan Basiri et al. [228]	SpamPer corpus, outcome reveals SVM are good enough by utilizing metadata and composed feature return a hopeful outcome when used on unbalanced version of the current dataset for categorizing type one spam, for type 2 spam decision tree classifiers is good comparison to SVM and NB.	For spam detection in Persian utilized supervised machine learning, Naive Bayes, Support vector machine, decision tree classifiers are utilized for categorization.
Elahe Rabbani et al. [121]	For the present technique, created the host's Persian dataset. The results reveal that identifying Persian website spam with the BOSW approach and SVM classifier has the maximum efficiency.	The Bag-of-Spam-Words (BOSW) algorithm was used to detect web spam in Persian.
Mohammadreza Smadi et al. [336]	The author provides the TAJ dataset and shows how the BERT-SLP and BERT-CNN models beat previous baselines and traditional machine learning methods.	BERT pre-trained model for detecting false news, BERT-SLP for embedding combination, and BERT-CNN for extracting new features from embedding tokens. For Persian false news identification.
Mohammad Mohsen Sadr et al. [332]	The system outperforms K-NN, random forest, Bayesian, linear regression, SVM, perceptron neural network, decision tree, AdaBoost probabilistic gradient, additional tree, gradient boost, with 92.05 percent performance ability, 91.10 percent recall, and 91.57 percent F1 score.	For detecting bogus news in Persian tweets, an LSTM neural network and a 14-Layer BLSTM neural network were used.

2.2.11 Speech Processing and Recognition

Hadi Veisi et al. [291] provide a deep learning-based strategy for Persian speech recognition. The author applies techniques to the Farsdat dataset. To extract characteristics from speech signals, Persian Acoustic Models (AM) are built using Deep Belief Networks (DBN). Persian voice recognition is made more effective by the Deep Bidirectional Long Short-Term Memory (DBLSTM) and Connectionist Temporal Classification (CTC) output layer. Features of DBN and DBLSTM performance were good when compared to the HMM model by 8.1% and the Karel-DNN model by 3.1%. By using bidirectional networks in shallow and deep structures, voice recognition accuracy is increased.

Meysam Mohamd Pour et al. [34] develop a strategy to improve the Persian language's automated voice recognition. The UTA algorithm is used in the suggested strategy to reduce the learning time from, 18,000 to 6,500 epochs. The present method has a medium accuracy of 98%. Take care of the existing approach's specification while maintaining outstanding performance with fewer training samples. Building a network that can do voice recognition on a large vocabulary is the author's goal.

Ghamarnaz Tadayon Tabrizi et al. [62] based on the HMM model, provide a method for Persian voice recognition that can identify children's speech. Using two models—one trained on children's speech and the other on adult speech, this method applies adaption strategies. Additionally, speaker normalizing techniques such as model adaption and vocal tract length normalization were applied. An investigation using an adult model demonstrates that when using an adaptation strategy, short performance may improve by 37%. These techniques have been shown to increase the identification rate by 7% when applied to a recognizer tuned to recognize children's speech.

Naveem Srinivasamurthy et al. [11] to solve this issue, provide a method for recognizing Persian speech with fewer Persian voice samples using a data-driven phoneme mapping strategy. When borrowing speech data from English speech data, the author frequently uses English as a high-resource language. Persian models are retrained using seed models created by data-driven sound mapping of English models for assessment. The outcome reveals a sound error rate of 19.80% inflected to a sound error rate of 20.35% when Persian models were trained again from seed models which are constructed from sparse Persian data.

Yousef Pourebrahim et al. [327] employed a unique auto-encoder to correct the discrepancy between the source and target data by employing joint feature representation. The method's goal is to distinguish between different languages' emotional speech. An experiment using five publicly accessible datasets demonstrates that the present system can recognize emotions more accurately than the corresponding transfer learning approach. Future writers frequently improve the existing system using a recursive deep neural network and a semi-supervised learning strategy. In comparison to cross corpus training, the proposed technique increased the rate of emotion identification in the Persian Emotional Speech Dataset (PESD) by 8% when the source training set is the EMOVO Corpus.

Mohammad Nazari et al. [26] utilized to handle the non-linear dimension reduction method and apply kernel-based characteristics for individualized speaker vowel recognition. To improve the accuracy of the existing method, nonlinear kernel data mapping and an SVM classifier are used. On the Farsdat database, the system's classification effectiveness was compared to that of other vowel recognition techniques in an experiment. The current system, KDDA+SVM, outperforms LDA, GDA, and pure SVM in vowel recognition with 96.1% in the training set and 93.9% in the test set.

Hossein sameti et al. [58] continue working on voice recognition systems while developing a big Nevisa vocabulary for the Persian language. Many benchmarks and the most recent methods for modeling speech and language use the current vocabulary. Model-based approaches like MllR, MAP, and PMC are used in the procedure. Robustness feature techniques include CMD, PCA, VTLN, and RCC. From speech-increasing techniques, spectral subtraction and modified forms of wiener filtering are applied. A new robustness technique called PC-PMC is also used in the current procedure. For testing, proving the effectiveness of, and optimizing the parameter of the present method in a job that requires noise just like one that does not. Applying much robustness approach, the skillfulness of the system is washed away, and the noisy status satisfies us. The system is a real-world result as well as a non-rivalrous ASR engine.

Seyed Muhammad Hossein Mousavi et al. [244] in the current study, features like Mel-Frequency Cepstrum Coefficients (MFCCs), Spectral Roll-off, Zero Crossing Rate, Spectral Centroid, and Entropy Energy are used to identify the musical instruments. The feature is derived from a unique audio signal database that compiles audio samples from seven pieces of traditional Persian musical equipment. Utilizing a Multi-Layer Neural Network (MLNN), the classification portion was carried out using fuzzy entropy measurement. The system is the first effort made to classify musical instruments in Persian. The outcome is encouraging.

Mohammad Savargiv et al. [138] describe a method for extracting the emotions of anger, sorrow, neutrality, happiness, surprise, disgust, and fear from Persian speech using an HMM-based classifier and at least one feature number. The experiment's findings indicate a greater medium accuracy of 79.50% in identifying the emotions described. The present system is 8.75 percent better than the previous one.

Niloofar Keshtiari et al. [132] present a database of emotional speech in Persian to aid in recognizing emotions in spoken Persian. 90 unique certified Persian phrases from the database are organized into five basic categories: disgust, fear, rage, sorrow, and happiness. 1,126 Persian speakers who are native speakers have accredited it. There are speeches in around 470 sentences. In a test of comprehension, 34 native speakers verified it. Speech recognition performance was more than five times the chance, 71.4 percent. The database is available to the public for use with voice and text emotion recognition.

M.H.Bokaei et al. [40] introduce Niusha, the first Persian speech-enabled IVR platform. The technique is a Persian text-to-speech component engine. The system's designs are easily adaptable to a range of fields, and new words may be added to the synthesizer due to use of the Voice XML (VXML) standard in implementation.

Omid Mohammad Nezami et al. [241] provide a database called ShEMO for Persian that can detect voice emotion. About 3,000 semi-natural remarks, or 3 hours and 25 minutes of chat data extracted from online radio plays, are included in the corpus. Five emotions—happiness, anger, fear, sorrow, and surprise—belonging to 87 Persian native speakers are included in the database. Pertaining to kappa score, 64% of the agreement among annotators is defined as considerable agreement. The experiment's results for gender are 58.2 percent, 59.4 percent for females, and 57.6 percent for males. For academic use, the current database is freely available. Amirhossein Tavanaei et al. [63] describe a method for Persian voice recognition using discrete Mel-scaled and wavelet transforms. The outcome for speech recognition using clean phonemes was 69.04 percent better than voice recognition using MFCC, which was 67.26 percent. When compared to MFCC-based voice recognition, the efficiency in noisy speech is 63.89%, which is 5.21% higher.

D. Darabian et al. [129] enhance the quality of Mel Frequency Central Coefficients (MFCC) accuracy if the background is noisy. For applying pre-processing on original noisy voice signals, exploit spectral mean normalization. The original pre-emphasized voice divides up into overlapping time frames. The modified hamming window is windowed. Next, higher-order coefficients of autocorrelation are brought out. Then lower order coefficient autocorrelation is removed. The first pass from the FFT block after the power spectrum of the result is summed. A Gaussian shape filter bank exploited the output. Logarithm and two compensator blocks: one means subtraction and the next one is a root block exploit to the output. The DCT transition comes last. For categorizing the output and judging the effectiveness of improved MFCC techniques. The updated method is more effective than the original one in a noisy backdrop, according to certain real-time voice recognition tests for varied tasks.

H.Veisi et al. [18] has done Vocal Tract Length Normalization (VTLN) extension and improvement using MILLER and MAP are investigated for HMM-based speaker autonomy and a vast vocabulary. To effectively recognize speech in a backdrop of noise in a real-world application. A line search-based approach is utilized to look up speaker-correlated wrapping factors. The parameters used in the signal spectrum to adjust speaker variation in vocal tract length. The MILLER was used to evaluate the complete adaption of the Gaussian mean and variance transformation. Regression tree-based taking on in a supervised manner is used in this method. The standard MAP utilized for speaker indemnification and environment changes. Together, these methods conducted experiments in four different noisy settings. The outcome was noteworthy in a noisy setting, since it allows the existing technology to be used in practical applications.

Seyed Mahdi Hoseini et al. [113] describe a method for Statistical Language model-based Persian voice recognition. The developers of the current approach use great adaption methods for Persian voice recognition. The test result demonstrates that the system improves recognition accuracy for continuous speech recognition.

Z. Ansari et al. [361] provide a strategy for speaker adaption based on eigenspace for Persian voice recognition. By segmenting the eigenspace according to model features, Enhance Eigenvoice (EV) in a large amount of data is achieved.

Mohamad Abdolahi et al. [15] use GARCH coefficients to perform a unique parametric speech modeling. For recognizing solitary digits 1 through 10 using GARCH Modeling characteristics. The results of the trial demonstrate a perfect improvement in recognition accuracy that outperforms Mel-Frequency Cepstrum Coefficients (MFFCC).

Mohammad Mohsen Goodarzi et al. [73] provide a method for improving the effectiveness of systems utilizing the GMM, GMM-NN, and NN techniques for rebuilding features in Persian telephone voice recognition. The famous technique for improving the audibility of speech that is band-limited was used by the author. The outcome of the experiment demonstrates that MFCC is superior for recognizing tasks. However, NN performance outperformed other reconstruction techniques and achieved the highest level of recognition accuracy. The author contends that the deployment of feature bandwidth requires a gender-based disjunction of speakers. Compared to telephone-matched models, the identification of gender is greater.

Ali Yazdani et al. [346] a deep neural network-based method for recognizing emotions in Persian speech. In this method, the Sharif Emotional Speech Database (ShEMO)[41], a database that was initially accessible in 2018, was used to test a variety of deep learning techniques. Using various deep neural networks and machine learning approaches, signal features in low- and high-level explanation, 65.20-percent unweighted accuracy, and 78.29-percent weighted accuracy are achieved.

Hossein Zeinali et al. [217] offer a database for speech recognition as well as dependent text, independent text, and text-prompted recognition for speakers. Deep Mine, the current database, is a sizable database for Persian and English. The most recent version has 360,000 recordings in total and more than 1,300 speakers. With the present Persian speaker identification database, researchers can create biometric voice applications for around 110 million users. The existing database may be used to train a potent Persian voice recognition model.

Seyed Mehdi Hoseini et al. [316] provide a method for identifying the emotions of rage and joy in Persian speech sounds. Additionally, the author created an appropriate Persian database for neutral and angry moods (with no emotion). There are 720 sentences in the database. Speech feather frequency signals acquired from Fourier transform like maximum, minimum, mean, and median LPC coefficients were also fetched. A Multilayer Perceptron (MLP) neural network is used to recognize the emotions of pleasure and rage. The final result demonstrates that the present algorithm is accurate to within 87.74%.

Z. Esmaileyan et al. [112] suggest PDREC, a new database for Persian emotion voice recognition. Radio drama series are the present database's data source. Additionally, the author offers a unique Speech Emotion Recognition (SER) system for Persian that simultaneously makes use of spectral and prosodic data. The Berlin database was used to assess the recommended technique. Additionally, the current SER was specifically created for men and women. Fisher Discriminant Ratio (FDR) was employed to eliminate characteristics that were not essential. The Linear Discriminant Analysis (LDA) embedding feature reduction approach is used to reduce the dimensionality of additional features. The samples arranged with LDA classifier. The consequences achieved 55.74% and 47.28% acquired for the actual database for males and females, hopefully. 78.65% and 74.40% accomplished aside the Berlin database for males and females, hopefully.

[41] https://github.com/pariajm/ShEMO

S.M. Ahadi et al. [19] enhanced MAP evaluated with triphone system to test the efficiency of MAP assessment with improved priors in development of triphone system utilized data-driven tying approach. It demonstrates how, in contrast to conventional ML-based parameter evaluation, the new approach may improve the performance of the triphone outcome system. The framework may also work well for tying other context-dependent modeling techniques.

H.Hasanabadi et al. [23] demonstrate a widespread and effective voice recognition system for spoken Persian words. Pattern classifiers and FFT-based features both benefit from MLP networks. The author suggested that a wheeled moving robot may be guided by verbal instructions in a setting that resembled a real house. The existing method may also be modified to use low-cost, straightforward microcontrollers. The technology is helpful for people with disabilities and is easily exploitable in many real-world situations. Additionally, it may be utilized as a standard composite speech recognition system.

Romina Oji et al. [326] provide a normalization open-source toolkit for speech recognition applications processing Persian text. The author considers Persian readable text to be numbered when it contains characters like #, currencies, , URL, symbols, date, time, phone number, and country code. The new technique outperforms other Persian text normalizing methods when the toolkit is compared. In comparison to HAZM, it performs better in text separation, and Parsivar exhibits commensurate efficiency. An experiment on the Persian Wikipedia dataset confirms the method's suitably high efficacy.

Arash Dehghani et al. [307] a time Frequency Convolutional Max out Neural Network (TFCMNN) is presented. It is a structure for side-by-side time-domain and frequency-domain 1D-CMNNs that are utilized concurrently and separately to the spectrogram, and then their results are consecutive and used jointly to a fully joint Max out the network for classification. To improve the performance of the present technique. Dropout, max-out, and weight normalization were taken from the book by the author. The FARSDAT dataset was utilized for the experiment by the inventor of the present system for performance assessment. The recognition score TFCMNN model in the present system evaluated with the 1D-CMNN model is approximately 1.6%, which is higher than the medium of traditional 1D-CMNN models. The system demonstrates time-frequency localization in automated speech recognition systems to boost training and improve system accuracy.

Mina Hamidi et al. [74] describe a neural network-based method for recognizing Persian emotions from voice. Due to the lack of accessible standard databases, the author created a database of emotional speech for the current approach. A total of 2,400 wave clips totaling these feelings—anger, disgust, sorrow, fear, ordinary emotions, and pleasure—are now included in the database. The pitch, intensity, and surrounding word characteristics of the voice signal were recovered by the author. In the end, using a neural network to automatically recognize emotions in speech, the result reveals medium accuracy was about 78 percent.

Mohammad Nazari et al. [20] provide a support vector machine-based method for speaker detection and grouping in Persian conversational voice. Mel-Frequency Cepstrum Coefficients (MFCCs) and MFCCs delta are used to train the suggested approach to assist vector machine classification. Using examples from the class, support vector machine training is applied to analyze the continuous speech signal of multi-speaker data in the Persian standard database FARSDAT. The outcome demonstrates that the present method, which employs a vector quantization-based classifier and the Gaussian Mixture Model GMM, provide high splitting quality [354].

Table 2.15: Persian Speech Processing and Recognition System

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Hadi Veisi et al. [291]	DBN and DBLSTM features are shown in the Farsdat Persian speech recognition dataset. Performance was superior to the HMM model by 8.1% and Karel-DNN by 3.1%; the accuracy of voice recognition will improve by applying bidirectional networks in shallow and deep structures.	Deep learning algorithms such as DBLSTM, CTC, DBN, HMM, and AM were used for Persian voice recognition.
Meysam Mohamd Pour et al. [34]	The results reveal that the present technique has a middling accuracy of 98% and great performance with Lex training samples.	The existing approach's learning time was reduced from 18,000 to 6,500 epochs using the UTA algorithm. To improve the performance of automated voice recognition in Persian.
Ghamarnaz Tadayon Tabrizi et al. [62]	The results demonstrate that employing the described strategy can improve recognition rates on children's speech by up to 7%. adults, on the other hand, benefit by 37% when adoption strategies are used.	The HMM model was used to train adult and child speech using vocal tract length normalization. For Persian language speech recognition in youngsters.
Naveem Srinivasamurthy et al. [11]	When Persian models are trained again using Saeed models constructed from sparse Persian data, the outcome reveals a sound error rate of 19.80% compared to a sound error rate of 20.35%.	For leveraging voice data from the English language, which is a high-resource language, a data-driven speech mapping approach was used.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Yousef Pourebrahim et al. [327]	The source training set is EMOVO, the recommended technique improved emotion detection rate in the Persian Emotional Speech Dataset (PESD) by 8% when compared to cross-corpus training, and the present system outperforms the transfer learning method.	For domain adoption, a novel auto encoder based on join feature representation and an unsupervised technique was used.
Mohammad Nazari et al. [26]	Dataset Farsdat The present system KDDA+SVM scores 96.1 percent in the training set and 93.9 percent in the test set, outperforming LDA, GDA, and pure SVM vowel identification approaches.	KDDA+SVM classifier was used to improve the accuracy of the existing technique. Vowel identification is important for Persian speakers.
Hossein sameti et al. [58]	The efficiency of the system is washed away by applying several robustness approaches, and the noisy scenario satisfies us. The system is both a real-world result and a non-rivalrous ASR engine.	Nevisa, a Persian language lexicon, was created to help with the Persian voice recognition system. Many common and cutting-edge talks and language modeling techniques were applied in current vocabulary. PC-PMC is utilized in noisy surroundings to assess the proficiency and optimize the parameter for clean circumstances.
Seyed Muhammad Hossein Mousavi et al. [244]	The outcome was promising, using a unique audio signal database compiled from audio samples of seven ancient Persian musical instruments.	To recognize the Persian classical musical equipment, features such as MFCCs Mel-Frequency Spectral Roll-off, zero-crossing rate, Spectral centroid, and entropy energy were employed.
Mohammad Savargiv et al. [138]	The results demonstrate a medium accuracy of 79.50% in recognizing anger, sorrow, neutral, happiness, surprise, disgust, and fear emotions in Persian speech. The result was an 8.75% performance improvement over the prior system.	For Persian voice emotion identification, an HMM-based classifier was utilized.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Niloofar Keshtiari et al. [132]	Speech recognition was 71.4 percent greater than five-time chance performance. The database is open to the public and may be used for text and speech emotion identification in Persian.	Created an emotion voice recognition database for the Persian language, which was validated in an understanding test by 34 native speakers.
M.H.Bokaei et al. [40]	Because of the use of the Voice XML (VXML) standard for implementation, the complete system may be easily changed in many domains.	IVR platform was used for Persian speech-enabled programs. Niusha is a Persian text-to-speech recognition system.
Omid Mohammad Nezami et al. [241]	The experiment's gender result is 58.2%, female individuals 59.4%, and male individuals 57.6%.	Created database for Speech emotion recognition by the name of ShEMO.
Amirhossein Tavanaei et al. [63]	Speech recognition in clean phonemes performed 69.04% better than MFCC-based voice recognition, which performed 67.26% better. In noisy speech, the efficiency is 63.89%, which is 5.21 percent higher than MFCC-based speech recognition.	For Persian voice recognition, the Mel-scaled discrete wavelet transform was used.
D. Darabian et al. [129]	In real-time voice recognition tests for diverse tasks, the modern algorithm outperforms the classic one in a noisy environment.	To improve the accuracy of Mel Frequency central coefficients (MFCC) when the backdrop is noisy. In order to perform preprocessing to the original noisy speech signal.
H.Veisi et al. [18]	The outcome was crucial in a noisy setting since it allows the existing technology to be used in real-world applications.	For HMM-based speaker autonomous massive vocabulary, MLLR and MAP are being investigated to expand and enhance Vocal Tract Length Normalization (VTLN).
Seyed Mahdi Hoseini et al. [113]	The results suggest that the system improves recognition accuracy in continuous speech recognition.	For Persian speech recognition, statistical language adaptation was used.
Z. Ansari et al. [361]	Enhance Eigenvoice (EV) in a huge number of data is acquired by separating the eigenspace based on model characteristics	Utilized Eigenspace-based speaker adaptation approach for Persian speech recognition.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mohamad Abdolahi et al. [15]	The outcome demonstrates a complete improvement in recognition accuracy, which is then Mel-Frequency Cepstrum Coefficients (MFFCC)	GARCH coefficient was used for identification of isolated digits 1 to 10 utilizing GARCH modeling characteristics.
Mohammad Mohsen Goodarzi et al. [73]	The results demonstrate that MFCC is the best for recognition, although NN outperformed the other reconstruction techniques and achieved the highest recognition accuracy. Gender-based recognition outperforms telephone-matched models.	GMM, GMM-NN, and NN approaches were used to reconstruct features, while MFCC and LFBE features were employed to increase the audibility of band-limited speech.
Ali Yazdani et al. [346]	Sharif Emotional Speech Database (ShEMO) yields 65.20% unweighted accuracy and 78.29% weighted accuracy.	Deep neural networks for emotion perception in Persian speech
Hossein Zeinali et al. [217]	Deep mining database is used to develop a sophisticated Persian voice recognition model.	For speakers and speech recognition, created a database for dependent text, independent text, and text-prompted recognition.
Seyed Mehdi Hoseini et al. [316]	Create a database for neutral and angry states in Persian (with no emotion). The present algorithm achieves an accuracy of 87.74%.	LPC coefficients were utilized to recognize happy and anger emotions in Persian audio signals, and an MLP neural network was employed to recognize happiness and anger emotions.
Z. Esmaileyan et al. [112]	Findings for males and females in the present database were 55.74% and 47.28%, respectively. The Berlin database obtained 78.65% and 74.40% for males and females, respectively.	The PDREC database was created, and the Berlin database was utilized for assessment.
S.M. Ahadi et al. [19]	In comparison to conventional ML-based parameter evaluation, the proposed technique improves the performance of the triphone outcome system. In addition, the system is employed in context-dependent modeling.	The effectiveness of MAP estimation with enhanced priors in the building of a triphone system employing a data-driven tying technique was estimated using improved MAP evaluated using a triphone system.

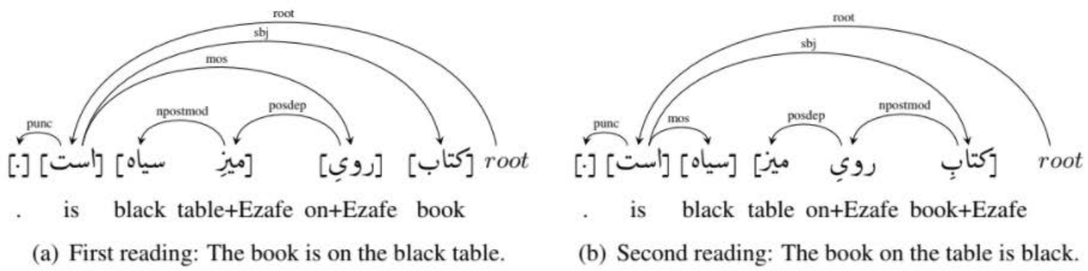
Paper	Outcome and Datasets	Preprocessing Phases and Approaches
H.Hasanabadi et al. [23]	The existing method may also be adapted to use low-cost and basic microcontrollers. Many real-world applications can benefit from the system. Specifically for those with disabilities. This can also serve as a standard composite speech recognition system.	In a realistic home-like setting, a wheeled moving robot was guided by verbal orders. The present technology is also extensible to low-cost and inexpensive microcontrollers, and it includes a voice recognition system for Persian lonely uttered phrases.
Romina Oji et al. [326]	The outcome in text separation compared to HAZM and Parsivar indicates adequate efficiency, and the experiment on Wikipedia dataset acknowledges the current method's suitable efficiency.	Created an open-source normalization toolbox for Persian text processing in speech recognition applications.
Arash Dehghani et al. [307]	The results suggest that time-frequency localization in automated speech recognition systems can improve system accuracy and speed up the training process.	The TFCMNN structure is an acoustic model's time-frequency localization capability. Based on CNNs, the suggested structure incorporates two concurrent time-domains and frequency-domain 1D-CMNNs, as well as a fully connected max out network.
Mina Hamidi et al. [74]	The results reveal that using a neural network for automatic emotional speech identification achieves a medium accuracy of about 78 percent.	Create an emotion recognition from voice database for the Persian language that comprises 2400 wave clips of 7 emotions.
Mohammad Nazari et al. [20]	The Farsdat dataset results reveal that the strategy is effective for separating quality by employing the Gaussian Mixture Model and a vector quantization-based classifier.	For speaker detection and clustering, a support vector machine was used for Persian conversation voice.
Shadi Pirhosseinloo et al. [82]	The Farsdat dataset results reveal that MPE-based DLT reduces word rate error more than MLLR adaption.	The discriminative linear for converting mean was evaluated using the Minimum Phone Error (MPE). Continuous voice recognition system in Persian.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mursal Dawodi et al. [265]	Isolated Dari Speech Database	Convolutional Neural Network (CNN) and Mel frequency coefficients (MFCC) are used to recognize individual Dari words after normalizing the sounds to a specified range from -1 to $+1$.

Shadi Pirhosseinloo et al. [82] provide a method for the Persian continuous voice recognition system for discriminative speakers, utilizing the minimal phone error (MPE) to assess the discriminative linear for transforming mean. The FARSDAT database was utilized to examine the effectiveness of the new technique, which showed improvement. As the last point, MPE-based DLT reduces word rate inaccuracy compared to MLLR adaption. Mursal Dawodi et al. [265] Describe a method for locating a lone Dari Afghan voice that uses a deep convolutional neural network. The author constructed a dataset because there weren't many Dari speech samples available. The dataset consists of 2753 utterances of 20 brief words, each lasting one second, made by different individuals, both male and female. The Dari language's single words may be automatically recognized using the Convolutional Neural Network (CNN). To understand how to represent features, it additionally uses Mel frequency coefficients (MFCC) during training. This model achieved a test set score of 88.2 percent. According to the results, the system can correctly identify samples of words that were used in training. However, it is trying to generalize terms outside the scope of training data and excessively noisy occurrences.

2.2.12 Parsing

Alireza Nourian et al. [136] describe the grandness of Ezafe building for betterment of the Persian Dependency tree and shallow parsing. Ezafe is like e at the end of the Persian word, for example: montazere (منتظری) (waiting Ezafe) naahar (ناهار) (lunch) which (منتظری ناهار) means in English waiting for lunch. Ezafe building data show off 4.6% improvement in dependency parsing, 9% improvement on relative shallow paring. In the experiment, the author annotated manually Ezafe tags in the Persian dependency tree-bank, which is a powerful feature for Persian syntactic parsing. In the example shown in 2.6 figure, two alternative Ezafe constructs are used to read the same phrase. Ezafe influences both phrase boundaries and dependence interactions, as demonstrated in the tree's [136].



root
sbj
mos
punc
npostmod
posdep
[.] [است] [سیاه] [میزِ] [رویِ] [کتاب] root
. is black table+Ezafe on+Ezafe book
root
sbj
mos
punc
posdep
npostmod
[.] [است] [سیاه] [میز رویِ کتابِ] root
. is black table on+Ezafe book+Ezafe

(a) First reading: The book is on the black table.

(b) Second reading: The book on the table is black.

Figure 2.6: Two alternatives Ezafe construct trees

Azadeh Kamal Ghalibaf et al. [31] describe a method for Persian semantic parsing based on standard features and a memory-based learning model system. The authors explain how employing a small training set of 1300 sentences might lead to improved outcomes. The author utilized a shallow syntactic parser with distinct syntactic units to break up sentences into two sections to extract the features. Both techniques make use of data from the hand-labeled Hamshahri corpus, which has the essential syntactic and semantic information. The outcome displays a 90.3 percent F1 score during the chunking stage and an overall Semantic Role Labeling (SRL) system score of 83.8 percent.

Mojgan Seraji et al. [139] introduced ParsPer, a graph-based parser and Mate Tools-based Dependency parser for Persian. The Uppsala Persian Dependency treebank's whole parsing representation, as determined by MaltParser, is used in the parser's training process. The treebank's method of syntactic annotation is based on Stanford Typed Dependencies with Persian enhancement. The experiment demonstrates a labeled accuracy rate of 82%. And 87% accuracy without labels. For Persian Parsing, the parser is openly accessible.

Mohammad Hossein Dehghan et al. [197] demonstrate a Persian-language phrase treebank structure. Nearly 30,000 sentences are present in the current treebank. The present treebank was created utilizing the bootstrapping approach. The tree is used to change a dependency form into a phrase form. The author created a phrase from treebank using sentences from the PerDT dataset. Dependency and Phrase forms, which are present in the current treebank, are two characterizations for the sentences. To train, model, and test the constituency parser, the author employed a fresh phrase from the treebank. For Persian NLP researchers, the current tree bank is accessible to the public.

Mohammad Sadegh Rasooli et al. [102]provide the Persian syntactic dependency tree's annotation phase for the linguistic attribute. Nearly 30,000 phrases with morphosyntactic traits and syntactic functions are included in the present treebank. The present tree bank has, 4800 unique verb lemmas in its sentence structure, making it a useful tool for research objectives. The present tree bank was created utilizing the bootstrapping method. The CoNLL dependency formatted data is publicly accessible to academics and is divided into the typical train, development, and test sets.

S. Lazemi et al. [235] based on the stacking technique, improved the Maximum Spanning Tree parser (MST

Parser) for the Persian language. MST Parser is suitable for this language due to Persian's special attributes. Due to the sharing of voice (SOV) characteristic, the head, and the dependent are typically placed far apart. Due to the creation of a sentence graph, MST Parser is therefore appropriate for identifying long-distance links. Expected to the unrestricted word-order characteristic of Persian, most sentences form non-projective trees, which MST Parser may also construct. Brown Clustering Algorithm (BCA) was initially employed for semantic characteristics to achieve the aim, and subsequently the stacking approach. they derived syntactic features from the Persian phrase-structure parser output. int the assessment used PerDT and Uppsala Persian dependency treebank. The consequence demos the description of new features to raise the performance of the dependency parser. The system gotten 89.17% and 88.96% on unlabeled attachment scores for PerDT and UPDT, respectively.

A.Pakzad et al. [154] provide a combined model based on Corpit software to improve the accuracy and feature optimization of the Persian and English languages. The authors use the Chinese language as a model for the Persian language. Dependency parsing and part-of-speech tagging on the Persian test data extend 85.59% for coarse-grained and 84.24% for fine-grained part-of-speech tagging for combined accuracy. A 76.01 percent for coarse-grained and 74.34 percent for fine-grained POS in English.

Soghra Lazemi et al. [236]The machine learning approach for the Persian language is the foundation of the suggested approach for semantic role labeling sentences. The technique extracts the feature from the dependency tree and uses it to classify verbal bench arguments. The current approach aims to improve the outcome by making use of clustering data. The experiment's findings demonstrate how appropriate and effective the existing approach is. And the system acquired a macro-average F1 score of 74.87 for the first corpus and 73.63 for the second corpus using the present strategy used for semantic role corpus in the Persian language. The corpora referenced in this work were made by the author.

Masood Ghayoomi et al. [71] to create an integrated genre using the Brown algorithm and word clustering approach to solve the data sparsity issue. The word's POS tags and derived attention were taken into consideration during clustering. This information was added by the author to improve clustering's functionality, particularly for homographs or ambiguity. The experiment's findings demonstrate the superiority of a class-based approach to a popular word-based parsing method. The statistical parser for Persian which was developed with a very little quantity of data now performs better using word clustering approach of parsing.

Mojgan Seraji et al. [85] display a Persian dependency treebank. The information is taken from the part of the speech dataset's Bijankhan annotated section. Utilizing a bootstrapping method with a dependency parser called the Malt Parser, an open-source dependency parser, and Stanford-like syntactic dependency annotation. The first experiment involves parsing, which is followed by training the parser on a seed dataset of 215 sentences that has been manually annotated with an accuracy of roughly 57% for the first interaction.

Parisa Saeedi et al. [84] Provide a feature set for the semantic role classification of argument instances in Persian. they investigate three key feature types in sentences that semantically identify chunks: chunks, verbs, and the relationship between them to encode argument instances. The verbs' valence structures are utilized to add generic and syntactical features to the feature set. The verb's argument categorization accuracy in the F1 test is 81.9 percent, up about 18.9 percent from the baseline accuracy. Using a small train set, they show that argument categorization can be done with accuracy.

Mohammad Bahrani et al. [45] provide a generalized phrase structure grammar (GPSG) model-based large-coverage computational grammar. The present model was developed for the continuous speech recognition (CSR) method, but it may also be used for other methods that need a syntactic analysis of Persian. The author examines numerous syntactic elements of the contemporary Persian language in the present methodology. The phrase structure model's report serves as an example. Basic syntactic categorization is indicated by the following symbols: verbs (V), nouns (N), adjectives (ADJ), adverbs (ADV), and prepositions (p). The X-bar theory contains four phases for defining nouns in accordance with the distinct issues in the Persian language's noun phrase structure. In the experiment, a bottom-up chart parser presses 100 phrases in Persian using the present grammatical model. The grammar model only takes 89 sentences into account. The word mistake rate for Persian continuous voice recognition decreased by 31% as a result of utilizing this grammar.

Roya Kabiri et al. [353]show the specific quality of Persian informal, and the lack of a specialized treebank for this option is caused by the lack of an annotated dataset for Persian casual. In the Universal Dependencies (UD) framework, the informal Persian Universal Dependency Treebank (iPerUDT) is interpreted in CoNLL-U format. Two parsers demonstrate that the very early trained on formal data encounter more unidentified tokens and structures when tested on informal data and lose the distribution of the basic patterns, generating an inaccurate prediction in such a status. The author shows that the majority of incorrect tokens and the dependence link, whose efficacy declines, are the unique attribute of informal Persian. The objective of this effort, which has a more significant impact, is to function as a stepping stone in showing the significance of informal language variances between languages, which have mostly been ignored by natural language processing technologies [354].

Table 2.16: Persian Parsing

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Alireza Nourian et al. [136]	When processing Ezafe Construction data, the result indicates a 4.6 percent improvement in dependency parsing and a 9 percent improvement in relative shallow parsing.	The significance of the Ezafe architecture in improving Persian dependency parsing.
Azadeh Kamal Ghalibaf et al. [31]	The Hamshahri corpus yielded a 90.33% F1 score in the chunking stage and an 83.83% total semantic role labeling system.	Used standard features and memory-based learning models. For the Persian Semantic role labeling system.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mojgan Seraji et al. [139]	ParsPer achieved an unlabeled accuracy of 87% and a labeled accuracy of 82%.	Mate Tools, Uppsala Treebank, MaltParser, and an upgraded version of the Stanford dependency parser were utilized for the current ParsPer parser for Persian.
Mohammad Hossein Deghgan et al. [197]	The PerDT dataset is used to convert a dependency tree to a phrase tree.	For the Persian language, the bootstrapping method was used to create a phrase treebank structure. The current tree has about 30,000 sentences.
Mohammad Sadegh Rasooli et al. [102]	The present treebank has 4800 unique verb lemmas in its sentence structure, making it a great resource for research purposes.	The bootstrapping approach was used to construct Persian syntactic dependency trees.
S. Lazemi et al. [235]	According to DAT and the Uppsala Persian dependency treebank, new features improve the parser's performance, which achieves 89.17% and 89.96% on unlabeled attachment scores for PerDT and UPDT, respectively.	Augmented MST Parser for the Persian language based on stacking method, Brown Clustering algorithm.
A.Pakzad et al. [154]	Dependency Parsing and POS tagging on Persian test data for combined accuracy range from 85% for coarse-grained POS tagging to 82.24% for fine-grained POS tagging, 76.01% for coarse-grained POS, and 74.34% for fine-grained POS in English.	Adopt a joint model of the Chinese language to the Persian language, for enhancing the accuracy and features optimization of Persian and English language based on Corpit software.
Soghra Lazemi et al. [236]	The result generated a Macro-average F1 score of 74.87 for the first dataset and 73.63 for the second dataset created by the author in the semantic role corpus.	The technique focuses on improving the outcome by clustering data. For semantic role labeling in Persian, a machine learning algorithm was used.
Masood Ghayoomi et al. [71]	The results suggest that a class-based method outperforms standard word-based parsing. The word clustering parsing approach improves the capability of the statistical parser for training with a little amount of data.	The word clustering approach and the Brown algorithm were employed to overcome data sparsity issues and incorporate genre.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Mojgan Seraji et al. [85]	Corpus Bijankhan. For the first round, train the Parser using an annotated manually seed dataset of 215 sentences with approximately 67% accuracy.	It is attached via a bootstrapping strategy that includes the Malt Parser, an open-source dependency parser, and Stanford-like syntactic dependency annotation.
Parisa Saeedi et al. [84]	The author looks at three key feature types in sentences that separate chunks semantically: chunks, verbs, and the relationship between them. The F1 score in the verb argument categorization is 81.9 percent, increasing the baseline accuracy by over 18.9 percent.	Provide a collection of features for semantic role classification of argument instances in Persian.
Mohammad Bahrani et al. [45]	Using this grammar in continuous voice recognition for the Persian language resulted in a 31% reduction in word mistake rate. It is also useful for other methodologies that need a Persian syntactic analysis.	The Generalized Phrase Structure Grammar (GPSG) paradigm was used to create a large coverage computational grammar.
Roya Kabiri et al. [353]	The concluding intention of this work, which has a significant impact, is to function as a stepping stone to uncovering the value of informal language variations, which have been largely disregarded in natural language processing tools across language	The CoNLL-U and the Universal Dependencies (UD) framework were used to understand the Treebank of Universal Dependencies (iPerUDT).
Mohammad Sadegh Rasooli et al. [283]	Create a new Persian universal treebank with 29,107 sentences, as opposed to the Seraji treebank, which has 5,997 sentences.	The PerDT dataset was transformed to universal dependencies using an automated method.
Kia Dashtipour et al. [262]	Corpora for Persian product and hotel reviews The present system outperforms state-of-the-art approaches. Logistic regression, SVM, DNN classifier, LSTM, and CNN had higher rates of 10-15% and 3-4%, respectively.	To improve polarity recognition, language rules and deep learning were combined. In Persian, this is known as concept-level sentiment analysis.

Paper	Outcome and Datasets	Preprocessing Phases and Approaches
Soghra Lazemi et al. [277]	Experiment for the first semantic role corpus in Persian language and the authors' given corpus produced F1 scores of 80.01 for the first corpus and 82.48 for the second corpus.	A dependency tree for Persian is used to build the semantic role labeling deep model.
Soheila Kiani et al. [32]	The experiment yielded a middling precision of 85.7% for the chunking result.	uTo chunk new sentences, a multilayer perceptron neural neutral with fuzzy C-means clustering was utilized. The Persian corpus utilizes rule-based tagging for automated chunking of the text.

Mohammad Sadegh Rasooli et al. [283] provide an automated method for converting PerDT to global dependencies. The approach also considers mapping dependencies, part-of-speech tagging, named entity identification, and tokenization. As a result, our solution does not rely on expensive and time-consuming human processes or blind one-to-one mapping. The experiment shows that learning a transfer delexicalized model with a total difference in label attachment score of more than 2% makes current annotations more compatible with universal standards. In the current method, authors contribute to fixing a variety of tagging issues in the original dataset, with the majority of these modifications being made manually and fixing annotation mistakes using the mapping pipeline. In contrast to the Seraji treebank's 5,997 sentences, the writers develop a brand-new Persian universal treebank with 29,107 phrases.

Kia Dashtipour et al. [262] provide a distinctive hybrid framework for concept-level sentiment analysis in Persian. Deep learning and language norms were used to improve polarity recognition. When a pattern is activated, the framework uses symbolic dependency links to allow sensations to flow from words to ideas. The framework switches to its subsymbolic counterpart and uses deep neural networks (DNN) to conduct classification when no pattern is triggered. The system outperforms cutting-edge techniques in experiments. Include convolutional neural networks, logistic regression, support vector machines, DNN classifiers, and extended short-term memories, with greater rates of 10-15% and 3-4%, respectively. Use Persian hotel and product review corpora.

Soghra Lazemi et al. [277] A dependency tree for Persian is used to construct the current semantic role labeling deep model. Every verbal predicate's possible argument is identified with dependency connectivity for each double-embedded predicate dependency path argument nominee using dependency tree information. To convert word characteristics into semantic role scores, the author employed a bidirectional recurrent neural network with large short-term memory portions. The experiment's results for the first semantic role corpus in Persian and the author's given corpus were 80.01 F1 scores for the first corpus and 82.48 F1 scores for the second corpus.

Soheila Kiani et al. [32] provide a hybrid method for automatically chunking Persian corpus text. To tag the corpus for neural network training, the present technique employs a rules-based approach. To partition new sentences, it then uses a multilayer perceptron neural network with fuzzy C-means clustering. The experiment's outcome demonstrates a medium precision for the chunking result of 85.7

Weston Feely et al. [111] provide the Farsi language's farsiNLPTools part-of-speech tagging, dependency parsing, and text pre-processing tools. Persian NLP researchers can get free access to the current tool online.

Mojgan Seraji et al. [104] provide a PrePer, or text pre-process, for the Persian language. For Persian NLP researchers, the current Persian text normalization is freely available.

2.2.13 Summary

The primary focus of our study is natural language processing in Persian, and this survey provides information on both traditional and contemporary methodologies. Although there are several datasets and corpora available to the public in electronic form, it was noted that there is still a lack of raw text. The crucial tools for text processing and creating an NLP framework are still in the planning or development stages. The development of tools and resources for Persian natural language processing systems must thus receive a lot of attention. By using the Cross-lingual or Pivot techniques, the majority of Persian language materials are created from wealthy languages like English. Components for Persian natural language processing may be created using the rich resource language research techniques. In this document, all works written in the Persian language are examined. This can assist academics and businesspeople in figuring out the viability, limitations, and breadth of NLP systems that employ Persian text processing.

2.3 NLIDB

Natural Language Interfaces to Databases (NLIDB) is one remarkable field that makes it easy to retrieve information from databases for non-expert users. The user can retrieve data without knowing SQL (Structured Query Language). In NLIDB, much research has been completed for different languages from early 1971 until current. The user can retrieve data in different languages using a natural language query (NLQ). NLIDB can work when users write a natural language query (NLQ) as an input. The system will recognize it to change NLQ to SQL queries after a few processes for accessing the database. For natural language queries like insert data, update data, delete data, etc. after the process is done, desired information will show on the graphical user interface (GUI) screen. In NLIDB, many approaches have been implemented, and each of them has a remarkable result. A database is a complete repository of data for retrieving information from it. Accessing data through Structured Query Language requires the knowledge of a domain expert. Working with SQL queries is difficult for non-expert users. For that, they are using NLIDB for retrieving data so that users can access data in their native language. NLIDB translates given sentences into SQL queries, but this task of translation is complex. Mr. Chomsky (1957) works on syntactic structures. BASEBALL (1957) was used for querying databases like LUNAR proposed by Wood et al. that achieved an accuracy of 95% and LADDER by Hendrix et al. in 1978 with an accuracy of 80-90%. There are numerous approaches done for developing NLIDB, like 1. Neural network 2. Syntax-based, 3. Semantic grammar 4. Pattern matching and systems for different languages apart from the English language. For neural networks, that is a forceful machine learning approach for NLP processing called Seq2Seq, a recurrent neural network. It is used for translating NLP to SQL and syntax-based use, a grammar that includes symbols and rules for grouping syntactically related words. Like grammar for parsing. And semantic grammar is similar to syntax-based systems, apart from non-terminal symbols, and the pattern is a work-based pattern and rule.

2.3.1 Test Databases

There are some databases which are often used as test databases for testing NLIDB systems. Some of these are described below.

Evaluation Benchmark ATIS database

A relational database called Air Travel Information Services (ATIS) contains data gleaned from the official Airline guide. The database has 28 tables and 125 columns in total. The ATIS0 Pilot corpus, designed by SRI in 1990 and consisting of 2,884 records, is the most commonly used corpus for querying ATIS for evaluating NLIDB queries[4].

Geobase Database

It is a Geographical Information System (GIS) database that contains data about the USA in the GIS domain. This database was used as a standard for testing the NLIDB system in the 1990s. The current database contains data about mountains, rivers, cities, roads, states, and lakes as well. Users can access to mentioned information through a natural language query and then the system changes it to the corresponding Structured Query Language. The SQL query access the database domain to retrieve data.

Northwind Database

This database is used to provide information about features in many of its products that were released by the Microsoft company like Microsoft SQL Server, Microsoft Access, etc. The current database contains 13 tables, about Northwind dealers company which import, and export foods globally. It includes clients, orders, stock, buying, providers, delivery, staff, and single-entry accounting is a great example of an ERP for a small firm. PostgreSQL is one of several non-Microsoft databases to which the Northwind database has later been migrated.

Spider Dataset

the Spider dataset[42] is a natural language query to the Structured Query Language dataset, with complicated and cross-domain queries for large-scale semantic research. The query corpus includes clauses and operations such as ("ORDER BY, GROUP BY, HAVING, SELECT, WHERE, LIMIT, JOIN, UNION, NOT IN OR, INTERSECT, EXCEPT, AND, EXISTS, LIKE, and subqueries"). Spider is a large NL-to-SQL corpus created to train the neural network utilized in natural language interface to databases. The Corpus includes 10,181 queries and 5,693 complicated SQL inquiries in 200 databases spanning 138 domains [253].

IMDb Database

The Internet Movie Database (IMDb) is a website that contains information about actors, directors, producers, movies, series, and television shows. Internet Movie Database is accessible publicly at [43]. Every dataset is housed in a zipped, UTF-8-encoded file with tab-separated values (TSV) formatting. Customers can access certain subsets of IMDb data for their personal and non-commercial purposes. Because the information is incompatible with the relational paradigm, various relational database versions with varying numbers of tables and columns have been produced.

Wiki SQL Database

Wiki SQL[44] is a corpus created by "Zhong, Xiong, and Socher (2017)" for training seq-to-SQL, a deep neural network-based NLIDB. Wiki SQL contains 80,654 annotated "NLQ and SQL queries involving 24,241 tables from Wikipedia". It is vital to note that the Wiki SQL corpus doesn't contain foreign keys and that an individual domain contains just one table [281].

2.3.2 Different NLIDB Related Systems

Expert System

F.Siasar djahantighi et al. [27] works on an expert system for recognizing word similarity in any language. By using a parse tree for parsing user input sentences, then transform or translate them to SQL query by using the

[42] https://yale-lily.github.io/spider

[43] https://www.imdb.com/interfaces/

[44] https://github.com/salesforce/WikiSQL

steps below, which are illustrated clearly.

A. Syntactic knowledge: Replacing data in the natural language process in the syntactic aspect contains these steps.

- Shallow parser: some function to give a point to the text. And give a superior score to the main structure, which is always used for an input sentence.
- Syntactic parsing. It is used to identify the grammatical roles of a word that is input by the user.
- Lexical cohesion: like one word in English, including different words that have their meaning according to the sentence that is used. For example, one word has many meanings in English and according to the sentences used.

B. Semantic knowledge: A natural language interface to a database needs to have a preprocess procedure to detect the changing of the word in different sentences, for example.

1. Who is (are) the author(s) of the book(s)?
2. Who is (are) the writer(s) of the book(s)?
3. Who is (are) the author(s) of the resource(S)?
4. Who is (are) the writer(s) of the resources(s)?

Find the similar word by the help of WordNet to build semantic database preprocessor, learn the database schema for recognizing entities and their attributes.

C: using of Prolog and Azmi in expert system. Prolog is programming in logic, which is used for natural language processing and AI. And this language is specific for answering the knowledge base question that has reality and structure rules. Azmi is a logic server and its interface for those programs is built by Prolog. This library is implemented in the C++ language.

D: Semantic Analysis: It is used to identify which word is related to which specific attribute or entity for mapping. For example, if the user types the NL query “List the address of the writer Mark Twain” in the first stage, the system created a parse tree to identify the grammatical rule and, after finding the exit meaning, then mapped to a corresponding entity or attribute in the database schema. **Output: SELECT address FROM author Attribute Relation, WHERE name = ’Mark Twain’;**.

E: System implementation: used MS-Access for the backend and VB.net for the front end. VB.Prolog and VB.NET. The author tends to create a knowledge base well-handled with the Persian language.

NADAQ NLIDB System

1. **Storage of data:** it made use of the MySQL database management system to obtain tables and information for training and translation models, as well as to execute SQL statements and display the results to the user.

2. **Model Administration:** it is the core of NLIDB, which employs various model two-way translations among plain language and Structured Query Language, in addition to avoiding needless user queries.
3. **User Interface:** it is the main component that sits between the user and the computer.
4. **Speech Recognition:** this section is used to convert voice to text using iFlytek.
5. **Translation:** this component's applications are based on a neural network coder-decoder for machine translation. By employing finite states for SQL parsing, it incorporates hidden states in the model; the hidden component is crucial for preventing erroneous output.
6. **Rejection:** this section allows NADAQ to disregard nonsensical user inquiries. In addition, it is utilized to determine user queries to the database.
7. **Recommendation:** this section improves the interface for candidate queries to the user for alteration, and selecting it aids the user who lacks SQL experience. Three databases were used to test NADAQ: Microsoft Academic Search (MAS), IMDb, and Geobase. Three techniques are compared in the test: neural network convolution machine, NLIDB, F1 score, semantic parsing model, and seq2seq machine translation scores 83.9-percent for Geobase and more than 80% for IMDb [252].

DBPal NLIDB System

this method employs a deep learning model to translate natural language queries into SQL queries and speeds up the translation process for wording variations.

Neural query translation: based on the seq-to-seq recurrent neural network paradigm for achieving quick query translation, NLIDB's key innovation is the synthetic generation technique. The training set builder embraces two phases: builder and support. The first phase displays NL-SQL pairings and a slot-filling dictionary capable of creating between 1 and 2 million pairs. The second one raises the starting size of NL-SQL Pairs automatically. To alter the NL component of the sub-pair automatically using many linguistic varieties.

Interactive auto-completion: it assists users with auto-completion tools and question recommendations when they are unfamiliar with the database schema. And assist them in writing sophisticated queries. DBPal for the patient and Geoquery benchmarks were respectively 75.93% and 48.95%.[293].

NaLIR System for Querying Relational Databases

It is an interactional NLIDB that illustrates to the user, gradually, how the interface understands the user's query. The interface displays a variety of explanations and interpretations for the user to choose from. This enables the resolution of ambiguity by interaction with the user. This system is made up of three parts. Which is discussed more below.

1. **Question interpretation:** this section contains a parse tree node mapper that interprets NLQ and represents the elucidations as a query tree.

2. **Communicator who interacts:** this section's role is to rectify the resultant interpretation for the user.

3. **Translation of a query tree:** this section's duty is to translate the query tree to a SQL statement and deliver the SQL query to a DBMS. Three modules make up the query interpretation component.

   (a) **Dependency Parser:** this step is used to create a parse tree from the input query using the Stanford parser. every node represents a word in the tree, so a linguistic connection is directly between two terms.

   (b) **Mapper:** structure adjuster parse tree receives the foremost mapping, and interactive communicator receives each candidate mapping from parse tree node mapper. Tokenizes and recognize mapped nodes into Structured Query Language components.

   (c) **Parse tree structure adjuster:** this component validates the parse tree's validity. If the tree is consistent with the database structure and the meaning is clear. This module is completed in two steps. First, the tree was reformed to be consistent with NLIDB's semantic coverage. Choose the best of multiple right candidate trees for the second stage. The selected parse tree is semantically assessed in this stage. The procedure is carried out under the control of the user.

   The system was tested with 98 inquiries from the "Microsoft Academic Search Dataset" and had an 89.79-percent accuracy on 88 accurate answers [118].

Natural Language Interface Layered Architecture Based

This method is built on layered architecture, since converting an NL query to a SQL query is a challenge. The SID (Dictionary of Semantic Information) is the foundation of the natural language interface to the database, storing the information required by the interface to process a query. The info can be stored as terms and phrases by a semantic information dictionary that relates to tables, column relationships between columns, alias values, and incorrect values allowing the essential data to be available and facilitating query interpretation.

1. **Lexical analysis:** this layer's job is to partition user requests into tokens, give POS tags to discovered words, and assess a possible search in the case of a not-found term in the lexicon.

2. **Syntactic analysis:** the shallow parser is related to this section to determine one syntactic class for terms with several classes, while avoiding extraneous terms.

3. **Semantic analysis:** this section conducts numerous activities for determining the input query is translated into Structured Query Language, which is added by the user. A layer of semantic analysis is made up of five child layers. To rectify inexact and assumed name values. These child layer senses, deal with

terms that signify inexact values, such as lunchtime, evening, and so on, aliases like noon, couple, fifth, or NY instead of NEW YORK. The method examines the input query and looks up the term in the semantic information dictionary (SID) to see whether it's declared incorrect or has an alias value.

4. **Recognition of tables and column:** it searches a word by semantic information dictionary to find what is related with a table or column. Identification of phrases like where and select. These heuristics sub-layer determines the inquiry segments that comprise phrase like where, select. All search value is assigned to a column based on the data type's proximity and similarity. The WHERE clause of the SQL is made up of the pair column-search value, and the leftover columns comprise the SELECT clause. Handling of congeries functions and groups. This child layer is used to detect and deal with query terms that pertain to aggregate functions and grouping. For instance, SQL query GROUP By is method is carried out by examining all word in the input question and searching SID to discover if the term is related to an aggregated function or grouping clause. For example, how many, first, smallest, minimal, maximal, biggest, best, etc. Values recorded in distinct columns may be of different types. Each column is related to a different word or phrase. Identifying the implicit connect when a connected graph does not satisfy search conditions comprised of a one table column and another table column, the SQL statement consists of a join condition; this child layer makes an attached graph by using heuristics that add the slightest number of joint conditions; once the attached graph is formed, the generations of the SQL statement are simple.

The ATIS database was used to test the system with 71 questions 90 percent accuracy was achieved. Pazos et al. 2016 configured with C-Phrase, ELF, and Geoquery250 standard the system, accuracies obtained, 56.4-percent for the NLIDB, 35% for ELF, and 54.4-percent for C-Phrase were obtained respectively.[213].

NALI System for Accessing Relational Database

The top-down authoring system employs an unannotated corpus of an example natural language queries to extract lexical phrases to make easy NLIDB settings. The notion of creating a configuration pattern without human involvement by utilizing knowledge from the database's structure and vocabulary is investigated by the bottom-up authoring architecture. The system uses English as its input language and SQL as its output language; it also believes that queries written in human language are free of grammatical and spelling errors. The syntactic parser's base is the symbolic method of decoding human language queries. There are four steps in the process of translating queries to SQL, which are described below:

1. **Lexical evaluation:** this step scans the inquiry words and conducts ("part of speech tagging, lemmatization, and named entity recognition") on the input query.

2. **Syntactic evaluation:** this step is used to create a parse tree from the input natural language query by help of Stanford dependency parser.

3. **Semantic evaluation:** is used to translate the query to an intermediate representation language that uses first-order logic to explain the meaning of the query.

4. **SQL conversion:** this step creates the SQL query based on the logical query.

Using Geoquery250, NALI was analyzed and achieved an accuracy of 74.5% and precision of 77.4%.[153].

NLI to Relationship Databases Based on Ontology

The interface's architecture enables the system's scalability and reliability. n-grams are used to achieve word sense disambiguation. The following six steps are used to convert natural language queries to SQL statements.

1. **Stop words removal:** this phase was used to remove a specified list of stop words from a natural language query.

2. **Stemming:** this phase's objective is to identify the root words of the remaining words. Content word extraction: Using a natural language toolkit, this step assigns POS tags to words.

3. **Syntactic Evaluation:** is responsible for processing the query via a top-down parser. Syntactic rules stated in Backus-Naur Forms are used to execute parsing.

4. **Semantic Evaluation:** using n-grams and an ontology built from the database schema, the ambiguity of a word's meaning is resolved.

5. **Formulation of a candidate inquiry:** using the EFFECN algorithm, execute natural language inquiries, table joins, and multiple table and column selections based on the search conditions stated in the query.

This NLIDB was tested using 100 queries to a customer database and achieved an accuracy of 84% and precision of 86% [163].

Dependency Parsing-Based Query Builder

In the Natural Language Interface to Database (NLIDB) part of speech tags, dependency types are extracted using dependency parsing. The dependency parse tree is a structured hierarchy of the root of a parent subdivided into children. Every node in a tree identifies one word, and node labels show the connection between parent and child. The process of natural language query to Structured query is illustrated below.

1. **Lexical assessment:** preprocessing of user queries is the task of this phase like tokenization, removing punctuation marks, and stop words from the user query which be added in the format of a natural language query into the system.

2. **Syntactic assessment:** This phase determined part of speech tagging, dependency type, and building a parse tree from the user query by utilizing the Stanford parser. The tags like adjectives, verbs, nouns, etc. are identifying the parse tree nodes pairwise.

3. **Semantic assessment:** The meaning of the query comes from the assessment of the dependency type with help of the current phase. Find the exact value from the table's column in the database, after mapping the nouns and tokens to the table attribute, columns, and find values. Joining tables attributes, finding values for creating the logic query.

4. **SQL query generation:** Finding correct values from the database table, and column to change user natural language query to Structured Query Language according to user input type like delete, update, insert, etc. The SQL query will build according to the natural language query that is typed by the user. The generated SQL query is access to the database domain for retrieving data to a user in the graphical user interface.

The reported accuracy for this NLIDB is complete 85% However, the standard used for the judgment [133].

Limited Natural Language Clinical DB Querying

The top-k technique is used in this approach to translate questions in limited to Structured Query Language (SQL). A special domain database that utilized a special domain language for retrieving data from the Clinical Information System (CIS) and analysis of the data in the clinical domain (CliniDAL). For the construction of this special domain NLIDB, one of the mentioned three models is utilized, Entity Attribute Value (EAB), Entity Relationship (ER), and "XML" is the initial building process of the system discusses below.

1. **Process of queries:** This component accepts a CliniDAL query as input and processes it by constructing a query parse tree utilizing its subcomponents (Parser, Categorized, and Optimizer). The query translator then goes through the parse tree.

2. **Query generator:** This component is made up of four subcomponents. Mapper is the initial subcomponent. To prepare information for automated mapping, the CliniDAL query tokens are mapped to "CIS database" tables and columns using the similarity-based method Top-k and certain NLO tools like tokenization, abbreviation expansion, and lemmatization. The translator subcomponent conducts two types of translations relating to the generic CIS data model: the CliniDal query is translated to SQL, and the query is converted to XML if the CIS stores an XML document. The Temporal Analyzer locates and maps the temporal entity database table corresponding to the query's mapped terms to the CIS model data element.

The NLIDB was tested using a clinical Information System database and a corpus of 108 queries, and it achieved an accuracy of more than 84%. [123].

Ask ME NLIDB System

It has a new feature: services for creating queries that decrease user learning time. Ask ME's architecture enables it to be automatically reconfigured for numerous domains while maintaining accuracy equivalent to ontology based NLIDBs. Ask ME is made up of two primary parts. Which are discussed further down.

1. **Ontology builder:** after connecting to a database, Ask ME searches the specific-domain storage for specific-domain references for every pair that has used the interface. If the database's ontology is not in the storage, the system takes information about "tables, columns, and relations" from the database schema to generate the ontology.

2. **Dynamic Parser generator:** this section generates the "lexical, syntactic, and semantic parsers and analyzers automatically", allowing natural language queries to be interpreted and translated into Structured Query Language for retrieving data from the database. This section has three subdivided parts, "the lexicon, the syntactic parser, and the semantic analyzer". For parsing, the dictionary employs the NLIDB's link grammar parser. Semantic templates are used by The semantic parser that is filled with the database ontology's ideas.

The ATIS database and 448 queries from the ATIS "Scoring set A" were used to assess Ask Me. The NLIDB obtained a 94.8 percent accuracy rate. [96].

Intelligent Relational Database Semantic Grammar System

This system's good quality is that it is simple to configure. Utilizing a lexicon and a collection of output rules, a semantic matching approach is used to convert a natural language query to SQL. Semantic sets for tables and columns are included in the dictionary. The NLIDB generates a SQL query, and the result is shown on the screen for the user. The Northwind database and the suppliers parts database were used to test this interface. The main goodness is built on semantic sets and rules that the database administrator can change. The system is divided into two modules, which are described further below.

1. **Preprocessor:** This module is generated automatically by the domain dictionary, as are the rules that are employed by a semantic parser. The conceptions are founded on "WordNet, the database design, and database administrator feedback", and the administrator may enter, edit, and remove rules.

2. **Processing runtime:** It employs the rule and attempts to match terms from the input query to preset "data structures, tables, and columns of the database schema", the rule demonstrates the relationship between the attributes of tables.

suppliers and components database, and the Northwind database were used to assess the NLIDB, and a number of five students were invited to write English queries for both databases. The query contained "40 inquiries for Northwind and 20 for suppliers" and components. The evaluation outcomes suggest that Northwind is 70% accurate, while Suppliers and Parts are 75% accurate [173].

nQuery NLIDB System

The interface can handle requests using aggregate functions, numerous criteria in the "WHERE clause, and phrases like ORDER BY and HAVING". The system was designed to work with the MySQL database management system. Before getting data from the database, the nQuery converts requests to SQL. This system is designed to retrieve data. Allow for the translation of additional data manipulation procedures. To decrease the complexity of database requests, MySQL may process insert, remove, and update interface translation requests. Taking input and translating it to SQL over various stages.

1. **Tokenizer and POS tags:** For tokenization and part of speech tagging of input, query utilized the NLTK toolkit.

2. **Examine tagged tokens:** This stage reads the tagged token and produces a noun map and verb list. Furthermore, the kind of SQL statement is determined: SELECT, INSERT, DELETE, and UPDATE.

3. **Map to attributes and table names:** The table set is generated in this section to reveal the table which contributes to producing the Structured Query Language (SQL) using the noun map and verb list. Since table names are often referred to in requests using "nouns and verbs", the noun map is utilized to select the table columns that are required in the SQL statement.

4. **Filter for redundancy and complete clause mapping:** Furthermore, superfluous tables and attributes are deleted through a filter technique.

5. **SQL structures:** SQL query template is selected according to the previous section determination. according to the previous section data like clauses' information, tables and columns, and map table attributes, the specified template is filled, and the final SQL query is generated for the user to access data.

nQuery was assessed using a corpus of "synthetic queries to a bank database and a university database", both of which had 11 and 6 tables, respectively. The NLIDB accuracy was evaluated using "75 queries for the university database and 50 requests for the bank database", and it achieved an accuracy of roughly 86% [186].

Aneesah NLIDB System

using the pattern matching technique, structure the NLIDB architecture. The system applies "a conversational agent based on a knowledge base, a scripting language, and a Structured Query Language query engine". The user interface has a modular building that allows for adaptability when querying a database of different domains. This is achieved by configuring the NLIDB. the section below illustrated the system in three steps.

1. **Constituent 1:** "a conversation manager, user interface, temporary memory, and a conversation agent" make up this component. It is formed by the "controller, pattern matching engine, pattern matching scripting language, and response analyzer". The controller establishes and executes an interaction with the user for directing the user in providing the information the user desires. This interaction is executed from scripts in the system knowledge base, a pattern-matching scripting language that allows conversations with the user.

2. **Constituent 2:** It includes a knowledge base that allows the current system to connect with the "sales history database" and can be set up to require access to an additional database.

3. **Constituent 3:** SQL query engine is included in this section, which is composed of a "SQL configurator, SQL executing, an SQL analysis". The responsibility of this section is to extract data from the database domain.

This NLIDB was examined using a "sales history database". For the assessment, they used two sets of contributors: group A comprised people without SQL knowledge, whereas group B comprised contributors proficient with SQL. The quality rate was 85.01-percent, and the overall accuracy was 92.96-percent [140].

Transfer-Learnable NLIDB System Based on Other Approaches

The technique follows the notion of separating data and database design while also accommodating the peculiarities and complexities of natural language. The technique entails removing natural language idiosyncrasies and concentrating on the "Structured Query Language (SQL) semantics queries" to create an independent domain and transportable NLIDB. The method is subdivided into three phases "conversion of a natural language query 'q' to its annotated form-a, usage of a sequence-to-sequence model for translating q-a to annotated SQL statements-a, and analysis of the results". In addition, the annotated SQL statement s-a is converted to a standard SQL query 's'. The annotation of the natural language query is conducted to recognize a word/phrase in the inquiry that may be used to refer to DB items "tables, columns, and values". However, the tokens used to refer to DB components vary according to the circumstance, and they are not always clearly mentioned "semantic ellipsis". The NLIDB makes use of database metadata information, including the database structure, database statistics for each column, and natural language phrases particular to a database, a column, and column values. The Wiki SQL dataset was used to train and assess the NLIDB, which comprises 87,673 natural language queries and their SQL translations, as well as 26,521 tables. The NLIDB was evaluated using the OVERNIGHT dataset after being trained with Wiki SQL and achieved an accuracy of 82%. The query accuracy for this test was 60% (a 22% decrease) [214].

Based on Semantic Representations and Ontologies, NLI

This NLIDB uses semantic representation to model the NLIDB's knowledge using the Ontology Web Language (OWL). The semantically modeled knowledge enables the system to deal with discourse (a series of connected queries) and query a database of various areas. The Customization Module is a critical component of this system. The NLIDB configuration is done in two stages, which are described below.

1. **Database Schema Extraction:** it pulls metadata from the database schema and identifies the pieces that comprise a relational database's structure, such as table and column names. The column data types and existing table relationships. To build the NLIDB Knowledge, the administrator merely needs to specify the connection settings to his or her relational database to the setup module. Following that, the configuration module represents the knowledge and saves it in an ontology, which is an .owl file.

2. **NLIDB Customization:** The customization module examines each of their names to construct the NLIDB's vocabulary, which is expanded with lemmas and synonyms. The customization module models all of these components using the recommended semantic representations. The configuration module lets the database administrator manage the knowledge created by the NLIDB, allowing them to add, delete, or update information. Furthermore, it permits the administrator to restrict the usage of superlatives. The configuration procedure comprises the following steps: Assign language terms to database tables and columns that may appear in user inquiries about tables and columns. Define superlative terms and indicate whether they pertain to a maximum or minimum value, as well as the columns to which the superlative can be applied. Indicate which database column has data that users might use as search values in natural language queries.

A class for identifying the vocabulary word was defined, as well as an object of attributes. A class was established to identify the vocabulary term, as well as an object attribute to link each word to the schema element. A class was also developed to represent the superlative term. Additionally, data attributes were established to identify the database columns on which the superlative may be used and to designate the aggregate function to be used (max or min). All the NLIDB's semantic information is created in this manner. The ontology created by the setup module is utilized by NLIDB to interpret natural language user queries as well as to generate the relevant SQL query, with which the information sought by the user is retrieved from his/her relational database. This NLIDB was assessed using the Geoquery250 benchmark, along with other NLIDBs (ELF, NLP-Reduce, and FREyA). Demonstrates good performance, with an accuracy of 85.2% [126].

NLI Compatibility With Knowledge Base

The approach is founded on the premise that a natural language expression (answerable question) has a one-to-one mapping to a formal query; hence, the natural language question is translated to a formal query by comparing it against NL expressions and selecting the most appropriate one. If the degree of confidence in this comparison is insufficient, the interface rejects the inquiry and does not deliver a response. The NLIDB converts a natural language inquiry to a formal query in two stages, which are described below.

1. **A knowledge base is used to generate system-interpretable expression:** During this step, all NL phrases are compared to the natural language inquiry. An ontology is used to produce NL expressions. The ontology was built in such a manner that each sub-graph of the ontology may be stated in natural language; each sub-graph can yield at least one NL statement. A sequence of NL tokens is created for each phrase to be compared against natural language inquiries. The sequence is referred to as normalized expression.

2. **The created expressions are used to translate user questions into formal queries:** a natural language inquiry is converted into a formal query during this stage. It is decided which normalized statement is comparable to the query. One or more normalized expressions are initially produced for a natural language query. Following that, pairs of normalized expressions are formed, with one element obtained from the knowledge base and the other from the inquiry. By picking the appropriate pair of normalized phrases, the main question was determined. The selected pair is then utilized to construct a formal query.

The NLIDB was tested utilizing Geoquery250, and it obtained an accuracy of 86.6 percent and a precision of 90.6 percent. [149].

Using a generative parser to translate queries

This NLIDB uses the database schema to generate a collection of candidate SQL queries, which are categorized in the creation phase using an SVM-ranker based on tree kernels. The system constructs a collection of SELECT, FROM, and WHERE clauses, as well as joins, using lexical dependencies and the database schema. In addition, clauses are joined using rules and a heuristic weighting process, resulting in a list of ordered candidate SQL statements. It is feasible to handle complex queries using nested SELECT statements recursively using this strategy. A re-ranker is used to reorganize the list of pairs of queries and possible SQL statements, each of which is represented as a syntactic tree. Ambiguity and mistakes may influence the interpretation of the user query; nevertheless, information from the database schema may be utilized to validate the validity of the chosen interpretation. The technique entails producing all potential SQL queries using information from the database structure for example, primary keys, foreign keys, data type, and so on, and then using a ranking mechanism to identify the most likely one. The NLIDB interprets natural language queries with nested SQL queries as well as complicated natural language inquiries with subordinate phrases, conjunctions, and negations. A technique based on correlations between lexical dependencies and SQL structure is utilized to generate a valid set of queries. And acquire a question's lexical relations by utilizing the Stanford dependency Parser, which returns the set of binary word correlations between a governor and a dependent (gov, dep). Where gov and dep signify a parent node and a child node in the parse tree, respectively. This NLIDB was analyzed with three Geoquery subsets, yielding an accuracy of 87.2%, precision of 82.8%, and F1-measure of 85% [72].

NLIDB system for languages other than English and Hindi

This interface accepts natural language input in the form of an inquiry or data manipulation action in Hindi. A semantic matching approach is used to convert this request to English. Following that, a semantically comparable Structured Query Language is produced from the English Request and submitted to a database management system, with the result displayed to the user in Hindi. The interface is an ontology based NLIDB that allows for "data access, insertion, updating, and deletion from a transport database". The method described below is used to convert a plain language request to SQL. Creation of a transportation database that maintains information about transportation services. Kind of request identification "SELECT, INSERT, DELETE, UPDATE, and aggregate functions". Mapping of Hindi question words to database components "tables and columns". SQL statements are generated by mapping input queries using database stored values. The SQL command is executed, and the result is shown in Hindi. Two databases are used in the current NLIDB system, "a compiler database and a transport database". "Tokenizer, Mapper, SQL Query Generator, and database administration system are additional significant elements in the system". The procedure followed in this section is described side by side. Tokenizers convert the input Hindi query into words or tokens, which are then stored in an array. These Hindi tokens are maintained in the dictionary (system lexicon) alongside their English similar words. The mapper compares the retrieved tokens with tokens stored in the lexicon sequentially (system lexicon). Where the mapping is done. Words that match the relevant English words are kept along with their kind, while the others consider worthless and eliminated. The SQL query was created using the table and column names. The produced SQL command access to the database domain to retrieve data from the user in the graphical user interface. This strategy has no testing results [180].

Arabic NLIDB System

It is an interface built to change Arabic NL query to Structured Query Language. It is independent of a particular area and, in its knowledge base, can learn from experience. The user can retrieve existing data from the database domain in the Arabic language. The system uses Natural Language Processing (NLP) methods to convert requests into SQL commands. The most significant benefit of this method is that "it is independent of the database's language, content, and model". An intermediate representation language, which transforms a natural language query into an XML logical query, is the foundation of how this interface works. Describing the formal query in the database's oriented, relational-object, or XML format. Arabic NLIDB lies in three modules. Which is elaborated below.

1. **Linguistic constituent:** This module analyzes the NLP (morphologically, syntactically, and semantically) and creates the formal meaning of the query in the expression of XML.

2. **Database knowledge constituent:** This compartment converts the formal inquiry generated by the first compartment into a SQL query. The Structured Query Language (SQL) command is delivered to the database domain, which generates the tabular output. the adaptation of language component and knowledge database components lets the current system natural language interface query different domain databases.

3. **Natural Language query definition:** This compartment assists the interface in reusing antecedent refined queries to save translation time.

A corpus of 1,300 synthetic queries was used to test the NLIDB. The interface accurately answered 1,166 questions, resulting in a 95.1-percent accuracy.[144].

NLIDB in Vietnam langauge

This NLIDB is made up of two major components: the Question Analysis module y result in the Computing Module. The first component determines and collects information from the kind of user inquiry. The second component recognizes the user's 'y' desired information and computes inquiry statistics. For people and businesses interested in economic data from surveys, this provides a domain-specific interface for 'a' survey database. This NLIDB is made up of two parts, which are detailed below.

1. **Question Analysis Constituent:** The goal of this component is to extract the question word, question type, and question information from a user query. The input query contains "special words and phrases" in addition to the requested information, the NLIDB employs "Java Annotation Patterns Engine (JAPE)". Question words are accurately recognized as words using rules that detect them and combine the findings into a Vietnamese word segmentation VNTokenizer. Many sorts of inquiries are in Vietnamese, including YES/NO, compute, offer Reason (why), and compare; thus, to address various types, the system uses a statistical technique for answering questions that satisfies the user's need for data stored in the database.

2. **Outcome computing constituent:** This section determines the database columns that relate to the user query's inquiry information. This system "analyzes survey questionnaires and organizes the information from each survey question"; this process is known as meaningful data retrieval. A dictionary of synonyms for the economic and statistical fields is part of this system.

A database of economic surveys was used to evaluate this NLIDB. The query corpus consists of 500 user-generated questions, 300 of which were used to train the system and 200 of which were used for evaluation. Out of the 200 questions, 157 were answered right, for a 78.5 percent accuracy. [80].

Commercially Available NLIDB: Access ELF

The system is used in the commercial domain to convert natural language queries into Structured Language Query for accessing the database and extracting data from the user. The system is considered one of the best which has positive effects on the business domain. The current system used Microsoft access for the backend database, and the system is not adoptable for giant databases. Numerous characteristics of the system are illustrated below.

1. **Domain individualism:** Once installed on a PC, ELF may communicate with any database after configuration of the interface.

2. **Automatic configuration:** The automatic setup looks for tables and columns in the word and utilizes them to generate the lexicon. These are known “as synonyms, and ELF uses them as references for tables and columns when they appear in natural language queries”. The automated setup also saves the data type and table name for each column. The arrangement is completed by assigning “new synonyms to tables and columns or by assigning synonyms to terms that have already been associated with tables and columns”.

3. **Database Semantics:** during the user query analysis, ELF analyzes the phrases (known as synonyms) used to describe database tables and columns, and it utilizes its lexicon to try to forecast the synonyms used in inquiries.

Conlon, and James (2004) provided an assessment that included human resource professionals as users. The claimed recall success rate was 70-80%, and ELF was tested using the Northwind database and reached 91% accuracy [156].

EasyAsk NLIDB System

Easy Ask is a commercially available NLIDB that is used for querying an e-commerce database; it is a software search tool that combines natural language technology (Quiri) and analysis. Users can use keywords or phrases to narrow their search results. This system delivers product ideas, which result in particular categories of items that correspond to the description of user requirements. This interface assists users in formulating inquiries by providing alternatives as they write their inquiries. Semantic processing, which connects distinct text descriptions to ideas, is one of the most significant elements. (Also, relax by avoiding unimportant or unfamiliar terms.) Quiri, an NL technology that blends language processing with data interpretation, is a key component of EasyAsk. Quiri breaks down a user inquiry into words, then arranges them into phrases and normalizes the material for interpretation. It also offers spell checking, stemming, and synonyms. EasyAsk was tested against the Northwind database and achieved an accuracy of 31% and precision of 48.4%. prototype interfaces that use natural language [364].

ATHENA NLIDB System

ATHENA is an IBM prototype NLIDB designed to perform complicated nested SQL queries for commercial applications. The system use domain ontologies to represent the semantic items and their connections in a domain. This method requires no user training or feedback. For handling a user inquiry with many nested inquiries. The following components are used in the interface:

1. **Evidence Annotator:** This component examines all the tokens in the natural language query for collecting the evidence of one or more ontology elements. The user inquiry mentioned ideas, attributes, and relationships between concepts. Entities in database tables and columns are tokens that are mapped to certain ontology items.

2. **Nested query Detector:** a reasoning submodule collects data from a language analyzer and semantic annotators in order to find a probable nested question.

3. **Subquery Construction:** in the case of a nested inquiry, this component uses lexicon-based techniques to separate the user question into two pieces, the first for the outer query and the second for the inner query. This component in particular uses the outputs from the annotator and the domain element to apply a set of rules to the query and identify the appropriate collection of tokens associated with each inquiry. This component generates two Ontology Query Language queries (OQL).

4. **Subquery Join Condition:** this component creates the join condition that involves the inner and outer queries for creating the entire query in OQL for a nested query. The creation of joins is dependent on language analysis and domain reasoning.

5. **Query Interpreter:** this component's job is to transform an OQL query into a SQL statement.

ATHENA was evaluated using the FIBEN dataset, which comprises realistic business intelligence queries and integrates data from two separate financial subdomains: the SFC (Securities and Exchange Commission) dataset and the TPoX (Trade Promotions and Exchange Commission) dataset (XML transaction processing benchmark). This test yielded no performance results (accuracy or precision) [160].

English Query Natural Language Interfaces Discontinued

Microsoft created an environment for establishing an English query model in their commercial software that a database administrator may utilize to put up an NLIDB. However, because databases differ and users ask a wide range of questions, building a model for answering user queries can be a difficult task. On top of the schema, a semantic abstraction layer is built, as well as a mapping between them. Entities represent database tables and columns. Allow English Query to convert a query to SQL in a model. Using the development environment, database administrators may define additional entities and relations. As an example, joining and shifting entity attributes such as related terms, entity type, field, and help text. English Query received a 70-80% evaluation; it was tested using the Northwind database and received an accuracy of 39% and precision of 46.1%.

SAA Language Access

This interface, created by IBM in 1990, is based on the usage of grammar and dictionaries to translate an NLQ to a SQL query. To that aim, it is made up of three major components, which are detailed below:

1. **Query user interface:** This component is built on Query management features, allowing users to create queries in NL. Furthermore, the system use these techniques to provide findings to the user.

2. **Natural Language Engine:** This module's job is to syntactically parse the user inquiry, determine its meaning, and provide proper responses. The following components are included in this component. Lexicons are used to store general and domain-specific terminology. Grammar for examining the structure of the user inquiry and ensuring the right grammatical form of a question. Semantics, for assisting in understanding the precise meaning of a natural language inquiry, such as selecting the appropriate meaning of a word having many meanings. Pragmatics is used to solve situation-dependent interpretations of user questions, such as recognizing pronoun references in the inquiry.

3. **Customization tools:** These tools are used to define the vocabulary that users use while creating inquiries. This tool is often used by the database administrator, who is familiar with the database structure and the vocabulary used in the organization by a particular user. It is specifically utilized to include in the vocabulary database-specific phrases and acronyms to be used in user inquiries for referring to database elements table and column.

In NLIDB, Woods et al. [87] The first method that introduced in the field of NLIDB was LUNAR which is allows English language to access a large database of LUNAR sample information, the approach was impressive, and it can easily handle 90% of request without Error [3] The 2nd method that is present in this field is the Lifer/Ladder system, which builds an interface of natural language to access extensive distributed databases. This method works with semantic grammar parsing skillfulness for input, and the query generated was yielded as input for the observed database. That worked with a simple query table and a query of multiple.

Hanane Bais et al. [225] suggested a method that renders an interface to enable a user to access data from the Extensible Markup Language (XML) database utilizing Arabic inquiries. The interface used changed from Arabic Natural Language Query (ANLQ) to Database Language Query (DBLQ). The current system is based on morphological syntactic and semantic conversation. The morphological tasks are to split up the natural query into tokens, the syntactic operation task is to build a parse tree by Context-Free Grammar (CFG). The semantic operation applied to acquire an abstract tree related to the Arabic query, the consequence shows 92% of ANLQ with the answer, 98% for DBLQ syntactically correct, 96.86% of XPath queries that are correctly generated match ANLQ.

Ani Anisyah et al. [222] suggested scheme based on Natural Language Interface to Database (NLIDB) for Decision Support Queries to operate the Indonesian language that has been built that can't operate decision support data queries. The system of translating imperative sentences into SQL queries is performed in two phases. The first phase is identifying the imperative sentence and parsing the sentence into a parsed tree that is aided by the PC-PATR syntactic parser. The second phase is processing the parse tree into an SQL query. The parse tree is analyzed to look for parts of SQL, then do object seeking for SQL parts to ontology. The objects found are arranged into SQL queries.

Maldonado et al. [237] planned system based on natural language interface to Database established on the dialog flow voice recognition and text conversion API which applied for the users who do not have the ability to retrieve information stored in databases. The challenge is restricted by the lack of understanding of the SELECT statement and the clauses that can make it up. If a user does not have adequate knowledge about the SELECT clause, the user is incapable of acquiring information from the database. This is why the need arises to render a mechanism to users that eases the interaction and data acquisition with the databases using a natural language. Users without education in database languages are capable of retrieving information with this interface; while experienced users will reduce query creation time.

Kailash Pati Mandala et al. [239] Introduce a system established on A novel Bengali Language Query Processing System (BLQPS) in the medical ontology, which is used for retrieving data from databases utilizing Bengali language to transform Natural Language (NL) query to Structured Query Language (SQL). To facilitate the use of the database for users who have no knowledge of SQL queries. The current system works based on scoring and pattern generation algorithm to formulate SQL queries from the semantic analysis. The proposed method using HTML, PHP, MySQL, and AVRO Bengali software has been used to design the web-based front-end interface. Hypertext Preprocessor (PHP) is a server-side scripting language that is used for the back end, and the database has been implemented in My SQL, which is a database development tool.

Chalermpol Tapsai et al. [342] suggested a method established on the natural language interface to the database for data retrieval and processing. The system was employed to change Thailand language sentences that were typed via the user as a natural language query to transform them into a Structured Query Language (SQL) for retrieving data from the database. The current system utilized an algorithm named "Ranking Trie" which is an aggregation of pattern parsing, ontology, and fuzzy system for bettering lexical analysis, semantic analysis, and output transformation process. The analysis of the meaning of complex query sentences at all levels, like words, sentences, and query conditions, has never been presented before studies. In the experiment, the following method shows an outstanding result, especially in Thailand, a non-segmentation language whose output shows an accuracy of 0.96, a precision of 0.99, recall of 0.97, and F-measure 0.98 which is superior.

HAO WU et al. [345] proposed an approach to provide an interface for human-computer interaction to a large database for retrieving and monitoring large databases. The current system is used for industrial processes and power systems. Which implementation costs are high. This approach worked based on dependency parsing and SQL parsing based on deep learning to control real-time control interface for database query interface, the KWECS method achieves an accuracy of 96.5% and a maximum response time of only 0.74.

Ursin Brunner et al. [302]suggested a scheme established on the value net: a natural language to SQL query. The system learns information from a database, which is used for natural language to SQL query systems that integrate values using the challenging Spider dataset. The main thought of the following approach is to use not only metadata information from the implicit database, but also information on the base data as input for our neural network architecture. In special, the author proposes a fresh architecture sketch to extract values from a user question. Come up with achievable value nominees which are not expressly mentioned in the question, and use a neural model based on an encoder-decoder architecture to synthesize the SQL query. The author assesses the model on the Spider dataset challenge using the executing accuracy metric, a more challenging metric than used by most associates of the challenge.

Swati Kale et al. [319] suggested a scheme based on an intelligent system for transforming natural language queries into SQL and its execution. The system works in which the user can increase sentences like “what is the salary of Jan.”. By utilizing data preprocessing or removing stop words and POS part of speech tagging, then the syntactic analysis is also done for checking grammatical errors. In the next step, every single word is mapped into an expedient object in the knowledge base. The meaning of all words is a union and fund of the meaning of an English sentence that was already added by the user and then translated to SQL query. The result will display on the screen like “Select Salary From emp where emp-name =’Jan’;”. The determination of accuracy for the presumption approach for Recall 88%, Accuracy 83.66%, Error rate 16.66%, and precision 91% was accomplished by the proposed system. Complex queries and nested queries need to be upgraded to the system, users can work in the Indian language, Marathi as well. It means a user can put the question in mentioned languages to get the result from the database using NLP.

Prasenjit Mukherjee et al. [324] suggested a system established on natural language query managing using an extended knowledge provider system that can deal with assertive, interrogative, imperative, compound, and complex type query sentences. Extraction of noun phrases is an issue in natural language query processing. Almost numerous times, determiner, preposition, and conjunction are prefixed to a noun phrase, and it’s a challenge to recognize the noun phrase with a prefix at the time of query processing. This system is designed to recognize these prefixes and extract exact nouns or noun phrases for natural language query processing. Without any manual intercession, this system used this algorithmic step, natural language query reads and validation, POS tagging, substring formation, and a combination. Remove repetition of combinations, wherever applicable, semantic analysis, create SQL, and generate a response from SQL. The proposed system is eligible to generate the response against natural language queries from the user and can be adopted in much knowledge management systems to enhance their performance in handling Natural language queries.

Harjit Singh et al. [340] projected system that can get input of sentences in the Punjabi language from the user and, after processing, change it to SQL query for regaining data from the database using natural language processing. This approach utilizes a pattern matching technique to make an SQL query from it. Punjabi is a low-resource language, and due to the lack of resources, it's more challenging to process Punjabi text to the corresponding English text. Given the system is built from scratch through these steps of user-flow processing, the first system takes the question in Punjabi, starts preprocessing by cleaning, and replaces non-noun from nouns, tokenization, and stemming on a given sentence. Furthermore, the process was done with data element finder common Nouns' operator symbols, translating tokens to English proper noun transliteration, Punjabi English dictionary, SQL preparation, Mapping to Meta-Data, and data-Accessing to the target database. In the end, the screen shows the result data to the users. The proposed system achieves 94.6% average Accuracy in generating Correct SQL Queries input Punjabi language Queries, and the FI-score was calculated as 0.98 for the system.

2.3.3 Systematic Literature Review

Introduction

Access to data must be swift and adaptable for today's jobs. It is beneficial to have easy access to data for purposes such as presenting data, identifying trends, quickly accessing information to make more informed decisions, focusing on the most pressing concerns, etc. Users require a user interface that is well-designed in order to access a dataset or database. An interface like that isn't always accessible. Users may find its user interface to be confusing when it is available. Users must learn alternative graphical tools or database browsers for flexible access to data if there is no user interface or if there is an insufficient user interface. Such as Adminer, DBComparer, EMS SQL Manager, Firebird, SQuirrel SQL, DB Browser for SQLite (DB4S, formerly SQLite Database Browser), DBeaver, DbVisualizer, HeidiSQL, FlySpeed SQL Query, TablePlus, Navicat, Beekeeper Studio, Oracle SQL Developer, dbForge Studio, Sequel Ace, Database .NET, etc.

Data from databases may be accessed by executing instructions in SQL (Structured Query Language), a formal language that computers can comprehend. It could be difficult to retrieve data using SQL without technical understanding. One of the things that could make it difficult to issue SQL commands is the necessity of understanding SQL syntax. In addition, users must understand the data in tables and the relationships among them in order to formulate SQL commands, since SQL is closely related to database systems. To solve this problem, several scholars have suggested adopting Natural Language (NL), instead of formal query languages, such as English, Hindi, Marathi, Bengali, Arabic, Spanish, or French. NL may act as the best user interface between a non-technical user and a computer program.[223].

Related Works

Early attempts at Natural Language Interface (NLI) included the "blocks world" virtual interface SHRDLU, developed at MIT between 1968 and 1970.[2]. Using plain English, the database interface Lunar allows users to obtain chemical analyses of Apollo-11 moon rocks [6]. Chat-80 was used to translate English questions into Prolog expressions, which were then compared to the Prolog database. The Chat-80 source code was widely shared and used as the basis for a number of other test-bed natural language interfaces. ELIZA was developed at MIT between 1964 and 1966 by Joseph Weizenbaum. It functioned by assessing user input and responding in accordance with scripts, mimicking a Rogerian psychotherapist. The doctor's script occasionally made a startlingly human-like connection while knowing very little about human thought or emotion. A demo is available online at the Logic Programming Associates (LPA) website, https://www.lpa.co.uk/. One of the few systems that support temporal inquiry is Janus [1], [367]. The Ubiquity add-on for Mozilla Firefox is a collection of quick and easy natural language commands that integrate internet services to provide users access to information about their current and other webpages. Online service WolframAlpha uses structured data to calculate answers to factual questions and delivers straight answers. The announcement was made by Stephen Wolfram in March 2009, and it was made public on May 15, 2009. [376]. Siri, the intelligent personal assistant, is included with iOS. To answer inquiries and give ideas, the application uses natural language processing. According to marketing materials, Siri can make dinner reservations while attempting to hail a cab, learns a user's preferences over time, and personalizes outcomes [375]. Other programs that accept natural language input include Ask.com, Braina, GNOME Do, Hakia, Lexxe, Pikimal, Powerset, Q-go, Yebol, and others. The most typical use of natural language interface (NLI) is in command and control systems that employ automated voice recognition. PBX/Voice mail system for transmitting voice mail to people with medical difficulties such as blindness, deafness, and physical constraints. With sign language, braille language, and voice-to-text, cellphones are used to call home in case of emergency with speech recognition and are also beneficial for developers [367]. Mahesh Chauhan et al. created a system that works by mapping to comparable Hindi words using a matcher. This enables the system to comprehend the semantic meaning of the Natural Language (NL) query terms that the user enters. The Structured Query Language (SQL) inquiry is then created based on the words in the user's NL query. The system interacts with SELECT, WHERE, UPDATE, and DELETE records in the maintained database. The Hindi sentences are parsed to guarantee thorough comprehension, which contributes to the development of the final query [110].

Rodolfo A. Pazos Rangel et al. offer a method for creating a Spanish natural language interface to a database using a lexical parser, a syntax checker, and a semantic analyzer. The system can handle the SELECT and WHERE clauses for database queries [9].

Ismael Esquivel Gámez et al. demonstrate how to handle Spanish statistical phrases turned into SQL functions, grouping clauses, and order clauses precisely. Databases in real-world applications have a broader range of properties for grouping and aggregation. For retrieving statistical data from supplied database tables, the system employs the COUNT function, FROM, WHERE, UNION, HAVING, SELECT, ORDER BY, and GROUP BY clauses. It converts the statistical expression-filled query from the end user into the proper SQL SELECT statement. The present system's research gap and limitations include that if an NL query does not contain statistical information, the query is regarded as invalid, and the system does not transform it to SQL query language. All the “invalid” questions were revealed to contain phrasing that may be utilized as function arguments, but did not obviously state attributes or processing [366].

Ashish Kumar et al. The user’s Hindi language inquiry is converted into a SQL query using semantic matching and a tokenizer. The system may be queried in languages other than English by the end user, such as Hindi and Telugu. The system may employ SELECT, UPDATE, DELETE, and INSERT clause queries to get data from the database in one of the stated languages [95].

Dat Tien Nguyen et al. introduce a survey database with a Vietnamese natural language interface. Our system’s two components are Question Analysis and Result Computing. The technique has yielded promising experimental results on numerous statistical challenges. Our approach obtains 78.5-percent accuracy from 157 current answers, 35 incorrect answers, and 8 non-segmented questions. Furthermore, by recommending important findings to users, the system can aid users in locating relevant information in the survey database. A system featuring a natural language interface for individuals and businesses seeking statistical data from economic surveys [81].

Maldonado et al. Pattern matching was described, in which patterns are defined using a few explicitly provided rules. The results of the natural language inquiry are then converted into those rules. Only when a pattern matches the specified user input may these interfaces generate a query. He also emphasized syntax-based user interfaces, in which the tree is eventually mapped once the natural language query is parsed. There are also semantic grammar interfaces, in which the database query is generated by parsing the input question, constructing a parse tree of the natural language query, and then providing logical meaning to the parse tree. Intermediate Representation Interfaces: These interfaces convert the query into a question with a natural language internal meaning. Because this architecture first converts a natural language question into a logical query intermediate by a series of analyses, the majority of NLIDBs nowadays are of this type: semantic, syntactic, and morphological. When employing speech recognition, only 74.19%, or 23 out of 31 tests, were right when conducting 31 Spanish questions using the microphone and the Spanish language. The voice recognition API has various shortcomings in detecting distinct words in the wrong instances. Lacking speech recognition, text questions can be entered straight into the text, just as they would be in a natural language. In this situation, the results were 100%, which means that all 31 requests were properly executed. The present system only supports the SELECT and FROM clauses with the COUNT operation for questioning the database in the Spanish language [238].

Shivani Jindal et al. based on tokenization, semantic analysis, and syntax analysis, created a Hindi natural language interface to a database. The prototype converts a Hindi query into a SQL query with an equivalent Hindi output. It accepts queries with different patterns as long as they query the same database. First, utilizing Karaka and language grammar, the query is syntactically analyzed. The query tokens are subsequently assigned to the database entities that comprise the SQL query. The user is shown the outcome. To get data from the database, the system employs SELECT, FROM, and WHERE SQL clauses [115].

Pradnya Borkar et al. offer a graphical user interface for relational databases that is rule-based. The system will first detect the query type, such as SELECT, UPDATE, DELETE, CREATE, or INSERT, as well as whether aggregation functions are included. The appropriate mapping of tokens with database values may be accomplished by extracting table and column information from the input Hindi phrases. In a more user-friendly dialogue-based approach, the user may input their Hindi query via a voice interface [362].

Rupesh Nichante et al.Based on the lexical analysis, the mapper creates a Hindi natural language interface to the transport database. For getting data from the database, the system can map words or tokens that include SELECT, UPDATE, DELETE, WHERE, and aggregate functions. The system may be queried using natural languages such as English, Hindi, Marathi, Gujarati, and so on, and the end user can read the results in the same language [181].

Hanane Bais et al. using graph theory-based approaches, users of the proposed system can see the database as a single table. The recommended systems have the benefit of being able to operate outside certain database domains and learn more via experience. The system creates SQL queries via morphological, syntactic, and semantic analysis phases. The author prefers to tackle complicated Arabic inquiries and improve the present system's performance. The application of a generic multi-language interface to the database is left for future development [191].

Hanane Bais et al. to operate, the proposed technique makes use of extensible markup language (XML) databases. It transforms the user's French Language Query (FLQ) requests into XPath. When a command is received, it searches the XML database for a suitable answer. The system is capable of handling numerous FLQs. The present system converts NL queries to SQL queries using morphological analysis, syntactic analysis, semantic analysis, and an XPath generator from a French dictionary. This interface has the advantage of functioning independently of the database domain and extending its knowledge base automatically as it acquires experience. Experiment findings show that the system's methodology can create XPath queries for numerous FLQs. We'll keep striving to solve more difficult situations in the future. Furthermore, the author intends to use the approaches presented in the paper to translate questions into other languages, such as Arabic [258].

Hishshah Ghassani et al. by determining the interrogative phrase's kind, which can be determined from the question word used, an interrogative sentence in the Indonesian language may be translated into SQL. Through Indonesian interrogative sentences, an ontology database, and syntactic parsing, the sentences changed to SQL queries for retrieving data from the database. There is a syntactic parser for Indonesian called PC-PATR. Context-Free Grammar (CFG) is the type of grammar used by PC-PATR. they may specify the language that is utilized in grammar, PC-PATR, and lexicon. The system works with WHERE, SELECT, FROM, and GROUP BY SQL clauses. For future work, the HAVING clause in a SQL query can be handled. Because the semantic type of attribute may be identified, the attribute description is unnecessary [198].

Hanane Bais et al. The proposed interface performs morphological, syntactic, and semantic analysis on an

Arabic natural language query (ANLQ) before converting it to a Database Language Query (DBLQ). One of the advantages of this interface is that it is independent of any program and improves automatically over time. Future work was indicated by the author as a solution to more difficult Arabic issues. Furthermore, the interface should be expanded so that it can answer inquiries in addition to natural languages. For 92.69 percent of ANLQs, the algorithm creates an XPath query, and 96.86 percent of successfully constructed XPath searches match ANLQs [226].

Mahdi Refaideen et al. based on natural language processing (NLP), syntax analysis, and semantic analysis, offer a natural language interface to the database for the Sinhala language, which is widely spoken in Sri Lanka. To obtain data from the database, the system uses SELECT, INSERT, UPDATE, and DELETE clause queries. The old version of the system only supports the SELECT clause query; the new version has been changed by the current paper's author [323].

Jigha Kapoor et al. create natural language interfaces to databases using dictionaries' token filters, parsers that map English words to the database, a query creator, and access to the database for retrieving data from the database. The Punjabi language interface to a database is an example of computer-human interaction. The proposed system allows users to enter patient or doctor-related inquiries in Punjabi and receive replies in that language. The system employs SQL clauses such as SELECT, WHERE, and FROM [368].

Kailash Pati Mandala et al. through this phase, provide a Bengali natural language interface to a database using query tokenization, part of speech tagging, pattern creation without token substitution, and semantic analysis. The SQL query is constructed. The database is accessed to retrieve the needed data from the medical domain. The problem with this proposed technique is that the time complexity grows dramatically as the number of unknown tokens increases. Comparison research with other systems of a similar kind using a time-complex amortized analysis approach is required to lower the time complexity of the present system. The system employs SQL clauses such as SELECT, WHERE, and FROM [240].

Harjit Singh et al. the system accepts a dynamically formed Punjabi question, generates a SQL query from it using pattern matching techniques, converts the query's data element tokens to real database objects, and connects to many databases to obtain the required data. It was put to the test using inquiries from three different domains. The test results show an average accuracy of 94.6 percent in producing valid SQL queries from Punjabi input questions. The F1 score of the system was 0.98. It takes a flexible-formatted Punjabi question, generates a SQL query from it using pattern matching techniques, converts the query's data element tokens to real database objects, and connects to several databases to obtain the required data. The system might be linked to any database domain without needing any alterations [341].

Khaled Nasser ElSayed et al. the proposed system would convert users' original Arabic queries, such as inquiries or imperative statements, into SQL commands in order to obtain responses from a holy Quran database. It will act as a bridge between users and the database, parsing and executing a few small morphological operations in line with a subset of Arabic context-free grammar rules. The system retrieves data from the database using the

SELECT, UPDATE, DELETE, INSERT, and WHERE clauses. The database will gradually grow to include more tables and attributes. The solution's technology will also be enhanced so that it can handle more advanced search requests and tie the answer. To an explanation of the surah (chapter of the holy Quran) and each ayah (verse of the holy Quran) significance in the holy Quran [130].

Chalermpol Tapsai et al. the proposed lexical analysis, semantic analysis, and output transformation processes have been improved. Through the implementation of a new algorithm called "Ranking Trie", which combines pattern parsing, ontology, and fuzzy systems to allow users to retrieve and process data under a variety of conditions and sentence patterns. The learning dataset included 3,868 Natural Language Query Sentences (NLQSs), which were utilized to gradually test and upgrade the model. In the test dataset, 500 NLQSs were utilized to evaluate the model. The results showed that Natural Language Processing for Data Retrieval and Processing (NLP-DRP) was capable of retrieving data, processing it, and providing results. This met user expectations and had accuracy, precision, recall, and F-measure values all more than 0.9. Outstanding results are especially visible in Thai, a non-segmented language where errors in the lexical analysis technique can easily develop due to word segmentation issues [343].

Silyan Arsov et al. a method for constructing question formulations in a confined form of Bulgarian is proposed. It is found in the complex theory of relational algebra. Within the context of the research, system tools for the formulation of the user's inquiry are introduced. A response generating method is proposed that includes a mechanism for direct data access as well as a semantic analysis of a user's inquiry. Algorithms for producing replies to questions containing various types of constructs have been suggested. The purpose is to investigate the system's error rate in connection to user qualifications, the kind of queries asked, and the length of those inquiries [12].

Xiaofeng Meng and Shan Wang et al. provide a unique strategy based on database semantics (SCM) to deal with Chinese natural language queries, which significantly enhances the system's usability. The findings of the experiments show that NChiql has great accuracy and usability [8].

Natural language Interface and Requirements

A natural language interface is a technology that allows users to interact with it using natural language or our language as humans. The user can contribute input in the form of voice using a speech recognition system, as well as written input using a keyboard, for obtaining replies to the appropriate question. A natural language user interface may be required for a multi-modal application. An example would be an interactive navigation tool that allows the user to interact with a graphic map by speaking commands and using a pointing device. A chatbot, for example, might be a component of a web application.A person with a physical constraint can only communicate by voice using a system that can be utilized by several users. Due to security and privacy concerns, each user must be personally approved. A person with a physical handicap, for example, may require input by speech, a single switch, eye tracking, or a combination of these. A person who is deaf or has a speech disability may require keyboard input. Users demand a variety of output methods or devices. A blind user, for example, would require speech output. A user who is deaf or has a speech problem may require graphically displayed text output. A blind or deaf user may require braille output. Text display settings on the system must be changed for users with low vision or learning disabilities. The system must employ terms that users, particularly those with learning or cognitive problems, can comprehend. Offer choices for resolving the issue if there is a known and controllable short group of acceptable alternatives. To be understood by people who have learning or cognitive challenges, spoken language must be pronounced correctly. Some users with learning disabilities require simultaneous oral and written textual information presentation. If a time limit is unavoidable, allow the user to extend it before it expires, erase it or adjust its length, or both. Provide instructions outlining the capabilities of the system [374].

Research Questions

The primary aims of our systematic review are to assess existing difficulties, find the best solutions, and assess how successfully individuals handle them. Create tools for constructing databases with natural language interfaces that are not in English (NLIDB). Furthermore, we've prepared a few questions for you below, and we're expecting to acquire the finest answers by checking authentic sources. The following are the questions:

Q1, why user use NLIDB?
Q2, For how many Languages, the NLIDB system is built?
Q3, What solutions are provided by NLIDB to human beings?
Q4, Which approaches are used in the non-English system NLIDB?

Papers Replicating Process

To begin the search strategy in this work, a manual search of journals and a systematic review of literature related to non-English NLIDBs were employed. Papers from 2001 to 2022 were used in this systematic review, and they were obtained for review from IEEE Xplore, ACM, Science Direct, Springer, and other relevant periodicals.

Results

they identified the following solutions after studying each study question.

1. RQ1, why user use NLIDB?

Natural Language Processing (NLP) approaches have been developed to bridge the gap between computer and human engagement. NLP is one of the most active fields of computer and human interaction today. NLP allows a machine and a person to interact without the need for a laborious method. NLP-based interfaces are used in interaction processes where computers interpret and change natural language text or speech. Researchers have been developing Natural Language Interface to Databases (NLIDB) solutions for numerous languages and underlying database formats since the 1970s. Without knowing Structured Query Language, NLIDB computers execute natural language queries and extract essential information from databases [325].

2. RQ2 For how many languages is the NLIDB system built?

HINDI language, Telugu language, Marathi language, Spanish language, Vietnamese language, Arabic language, French language, Indonesian language, Sinhala language, Punjabi language, Bengali language, Thai language, and Bulgarian language are some languages spoken. For the user who does not know SQL queries, they had an NLIDB system that translated natural language into SQL queries for accessing data from a database.

3. RQ3 What solutions are provided by NLIDB to human beings?

The natural language interface to the database is an interface that allows the user to enter data in his or her native language in the form of text or speech. The NLIDB assists unskilled users who are unfamiliar with Structured Query Language in effectively retrieving data. Business is another part of NLIDB. By assisting them, they would be able to simply install their enterprises all over the world. For non-English native speakers across the world, the sole barrier to doing business is the lack of knowledge of SQL queries. Researchers, governments, hospitals, NGOs, and others may all profit from this. NLIDB is more suited for occasional users since there is no requirement for the user to devote time to learning the system's communication language. As a result, NLIDB assists firms in saving money on staff training to grasp how to deal with SQL. Learning is also a hurdle since understanding the SQL query takes time [6].

4. RQ4, Which approaches are used in the non-English system NLIDB?

According to the literature, the following techniques have been used to convert NL queries to SQL queries.

- Mapping to corresponding Hindi words using a matcher and semantic analysis
- Lexical parser, a syntax checker, and a semantic analyzer
- Semantic matching and tokenizer
- The pattern is matched to the given user input, Syntax-based interfaces
- Semantic, syntactic, and morphological analysis
- Tokenization, semantic analysis, and syntax analysis
- Lexical analysis, and mapper

- Graph theory-based techniques: morphological analysis, syntactic analysis, and semantic analysis
- The French dictionary as morphological analysis, syntactic analysis, semantic analysis, and XPath generator
- Ontology database, syntactic parsing,
- Arabic dictionary for morphological, syntactic, and semantic analysis
- Syntax analysis and semantic analysis
- Tokens filters by dictionaries, parsers by mapping English words with the database, a query creator
- Query tokenization, part of speech tagging, pattern generation
- Without replacement of token position, and semantic analysis
- Pattern matching techniques and Punjabi dictionary
- Performing parsing and a few minor morphological operations in accordance with a subset of Arabic context-free grammar rules
- Pattern parsing, ontology, fuzzy systems, semantic analysis, and lexical analysis
- Semantic analysis of a user's query is suggested and relational algebra

As a result, the majority of systems employed a dictionary to convert non-English terms to English words in order to convert Natural Language queries into Structured Query Language.

Comparison Analysis of different approaches

The Table 2.17 presents a comparison of alternative non-English NLIDB techniques and their performance [356].

Table 2.17: Comparison of different methods

Methods	SELECT, WHERE, UPDATE, DELETE clause	COUNT, aggregate Function	HAVING clause	GROUP BY clause	ORDER BY clause	INSERT clause	Performance
Mapping, semantic analysis[110]	✓	✗	✗	✗	✗	✗	Not measured
Lexical, syntax and semantic analysis [9]	✓	✗	✗	✗	✗	✗	Not measured
Statistical Expressions [366]	✓	✓	✓	✓	✓	✗	76.46%
Semantic matching and tokenizer [95]	✓	✗	✗	✗	✗	✓	Not measured
Question analysis and result computing. [81]		✓					78.5%
Semantic, syntactic, and morphological analysis.[238]	✓	✓	✗	✗	✗	✗	74.19% speech input and 100% in text input
Semantic, syntax analysis [115]	✓	✗	✗	✗	✗	✗	Not measured
Rule-based [362]	✓	✓	✗	✗	✗	✓	Not measured
Lexical analysis[181]	✓	✓	✗	✗	✗	✗	Not measured
Morphological, semantic, syntax analysis[191]	✓	✓	✗	✗	✗	✗	92.4%
Morphological, semantic, syntax analysis[258]	✓	✓	✗	✗	✗	✗	97.2% first test, 99.27% in second test
Question analyzer[198]	✓	✓	✗	✓	✗	✗	Not measured
Morphological, syntactic, and semantic analysis[226]	✓	✓	✗	✗	✗	✗	96.86
Syntax and semantic analysis[323]	✓	✗	✗	✗	✗	✓	Not measured
Lexical analysis, parser[368]	✓	✗	✗	✗	✗	✓	Not Measured
Scoring and pattern generation[240]	✓	✗	✗	✗	✗	✗	Not measured
Translation to English and mapper[341]	✓	✗	✗	✗	✗	✗	98%
Lexical analyzer, parser, and translator[130]	✓	✗	✗	✗	✗	✓	Not measured
Lexical and semantic analysis, fuzzy System[343]							98%
Theory of relational algebra system [12]	✗	✓	✗	✗	✗	✗	Not measured
Database semantics (SCM)[8]							Not measured

Summary

Users may benefit from using NLIDB to access databases. In this systematic review focuses on non-English NLIDB techniques, we examined over fifty published articles from IEEE Xplore, ScienceDirect, ACM, and Springer Link. We then went through the papers to find issues and provide future researchers in this field with an understanding of the present status of the research. The bulk of non-English NLIDB systems employed a semantic matching and dictionary-based technique to convert NL queries to Structured Query Language. The research study conducted using a current technique for non-English systems could be useful to researchers who seek to create NLIDB systems for other low-resource languages.

To improve user experience, NLIDB system usability and accessibility require much more effort. There has not been enough study done on large-scale surveys or assessing the efficacy of various NLIDB projects. Non-text and non-verbal input is considerably lacking in earlier research and might be a promising topic for future studies, such as sign language, braille, Brain-Computer Interface (BCI), and so on.

Chapter 3

Methodology

The spoken natural language query interface to the database is built to change the Afghan natural language query to a Structured Query Language for accessing the database. The system works based on semantic matching to change the user input sentences to SQL queries. The process of changing NLQ to SQL query is elaborated as follows, step by step. In the first phase, the input sentence is translated into corresponding English sentences with the help of Google Translate. The lexical analysis implements lowercase conversion, removing punctuation, and tokenization as extracted from input text. And the second phase syntax analysis is implemented for removing stop words as part of speech (POS) tagging, and stemming/lemmatization. In the third phase, we use a system dictionary that contains Persian words with corresponding English words, and the users add NLQ queries with the help of the system dictionary to know when the user wants to insert or delete records. Finally, mappers to the database column and row for accessing data to the database for retrieving desired data from the database.

3.1 Overview

The figure 3.1 show the architecture of Persian Natural language interface to database using lexical analysis, syntax analysis, semantic analysis for translation of Afghan language spoken sentences into equivalent SQL statement [355].

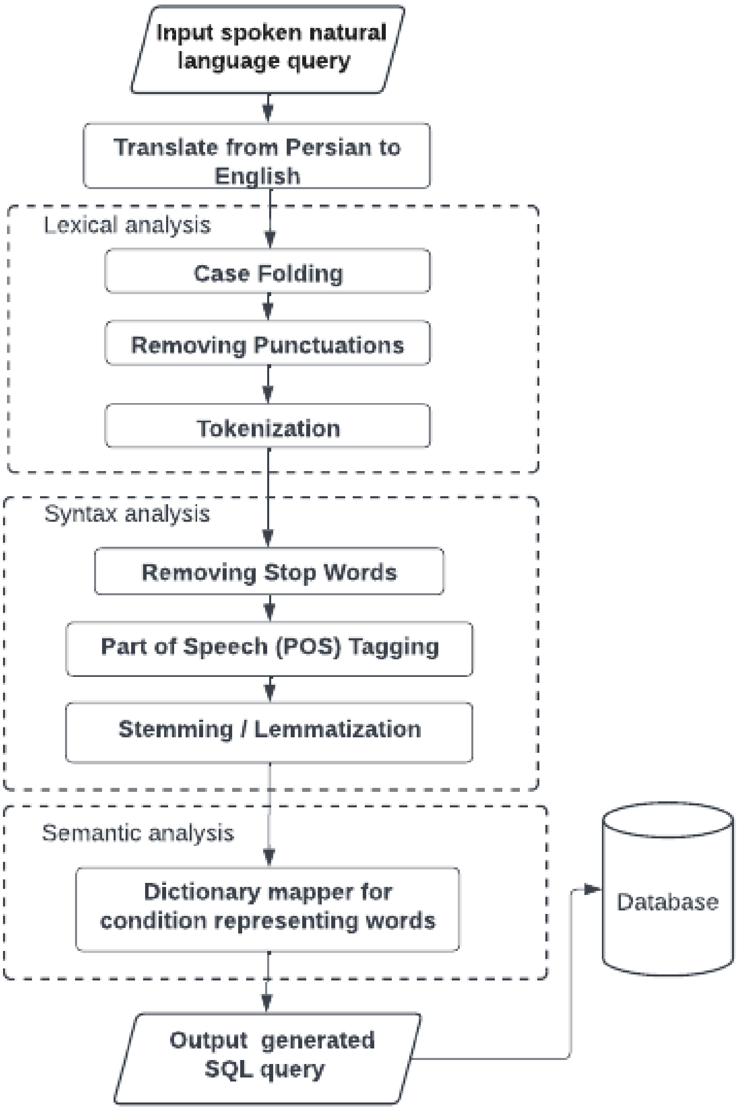

Input spoken natural language query
Translate from Persian to English
Lexical analysis
Case Folding
Removing Punctuations
Tokenization
Syntax analysis
Removing Stop Words
Part of Speech (POS) Tagging
Stemming / Lemmatization
Semantic analysis
Dictionary mapper for condition representing words
Database
Output generated SQL query

Figure 3.1: The architecture of the system

3.2 Natural Language Querying

An interface to the database is given in the Afghan (Dari) spoken natural language. The user puts a query in the Afghan language without any special syntax or format from a relational database. According to the data, the user wants to retrieve, the user always wants to find better ways to query databases in NLP to make their lives easier. Natural language systems try to build approaches that understand and respond to voice and text data and respond to text or speech like humans. Nowadays, systems, especially computer programs, used NLP for translating a text from a different language to a different language and also, using different areas. As text summarization is a tedious job for a human being, sometimes it takes a lot of time for people to summarize and concise one article or corpus, but by using NLP, it is very easy and efficient to concise a big Corpus. For example, voice-operated GPS digital assistants, speech-to-text, and chatbots can help the organization to simplify mission-critical business processes. The speech recognition task of NLP is used to send a human voice and convert it to text. Another task of NLP is speech tagging, which is used for grammatical tagging that tags every individual word of sentences into equivalent grammar like which part of the sentence is a verb and which part of the sentence is a noun. Word sense disambiguation: one word has multiple meanings, but the process of semantic analysis will sense the actual meaning of the work and distinguish between the sense of the same word and different sentences. Name entity recognition is used to identify New York as the name of the location, and Fred is the name of the man. Co-reference resolution is used to distinguish two words from the same entity, for example, a bear isn't an animal Large hairy person. Sentiment analysis is used to extract emotions Sarcasm confusion suspicion from the text. Natural language generation sometimes describes the opposite of speech to text, the task of adding structure information to a human language. In the Python programming language, there are numerous tools for NLP, for example, NLTK consists of many libraries for doing a specific NLP task. For example, sentence parsing, word segmentation, stemming, lemmatization, and tokenization for breaking phrases into tokens and passing them into tokens for better understanding of the text.

Gensim: It is an unsupervised and open-source library for topic Modelling and natural language processing using machine learning models and statistics. It is implemented on Python and Cython for performance. It's good for handling large text collection, and information retrieval. Gensim is also used to change Persian documents to vectors, and it's also used to train Word2Vec in the Persian corpus for finding word similarity. Gensim is being utilized for preprocessing and training Corpus to vectors very fast and efficiently.

spaCy: It is advanced natural language processing, an open-source library used in Python and Cython under an MIT license. The spaCy also supports deep learning, allowing linking statistical models which are trained with the famous machine learning libraries like PyTorch TensorFlow MXnet. The spaCy feature is a convolutional neural network model for dependency parsing, part-of-speech tagging, text categorization, and named entity recognition which supports 17 languages like Portuguese English Russian Chinese and also multilingual named entity recognition model and also supports 65 languages for tokenization.

Textblob: Simplified text processing is a library for textual data processing. In NLP, we used for part of speech tagging and noun phrase extraction sentiment analysis, translation, and classification. Textblob has many features like classification, Naive Bayes decision tree, word and phrase frequencies, and parsing n-grams. Spelling correction WordNet integration is very helpful for the Persian language too.

CoreNLP: It is one of the stop shops for natural language using Java that allows users to annotate linguistics. The text includes sentence boundaries, tokens part of speech tagging, named entities numeric, time values dependency, and constituency parser. Offenses sentiment quote attribution and relation Core NLP support 8 languages: Chinese, English, French, Arabic, German, Hungary, Italian, and Spanish. Core NLP is employed for producing pipelines like tokenizing raw text split and part of speech tagging lemmatization named entity recognition and dependency parsing, and the final output is annotations and final with a built-in sense for a sentence.

Hazm: NLP tools are used especially in text cleaning, word tokenizing, lemmatization, part of speech tagging, shallow Parser, Persian dependency parser, and interface for Persian corpora. It is compatible with NLTK, which is very helpful for researchers to preprocess and build a Pipeline to process a Persian Corpus.

3.3 Translating From Persian to English

Nowadays, Google Translate is a service that can translate 90 languages, it can translate phrases, words, text sections, and web pages. Which provides the best translation pattern among translated texts and human. At the start, Google Translate was a rule-based machine translation and after using a statistical model, it could translate words in 2006. SMT is multilingual corpora that contain a database of the sentences for the source language as well as for the target language. For a translation of Persian to English, this large group of sentences will provide for the machine to calculate the probability of the word. If the probability of X is 75%, then translate to Y. The most widely used and famous machine translation service in the world offered by Google is called Google Translate. To examine the acquired data, Persian-English translation mistakes and English-Persian translation frequency statistics were tabulated and compared. The incorrect "token translations, tense errors, wrong use of prepositions, word ordered error, errors in the distribution and use of verb group, active voice, and passive voice errors". It is juxtaposed in a separate table, and frequencies of different types of errors produced by Google Translate are tabulated for comparison. The overall frequencies of each sort of error with Google Translate for English-Persian and Persian-English Mistake translation were put in a table. A chi-square test was done to determine the difference between English-Persian and Persian-English poor translation. This part was about "the quality of Google Translate as a machine transaction for Persian to English or English to Persian". It shows Google Translate is convincing to use in Persian-English [147] [355].

Table 3.1: Google Translate error oftenest English-Persian translation in both directions

Error Types	Google Translate English to Persian	Google Translate Persian to English	Total
Lexiocsemantic	42	26	68
Tense	17	8	25
Preposition	15	5	29
Word Order	31	5	36
Distribution and use of Verb Group	18	9	27
Active and Passive Voice	2	2	4
total	125	55	180

Table 3.2: Google Translate's Chi-Square findings for comparing English-to-Persian and Persian-to-English translation mistakes

		Value	Df	sig.(2-tailed)
Pearson	Chi-Square	7.72	5	.172
Likelihood	Ratio	8.34	5	.138
Linear-by-linear	Association	1.72	1	.189
N of Valid Cases,		180		

Table 3.3: Translation utilizing different services

Translated by Duckduckgo / Microsoft Bing Translate	
نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۳ سال است	Show the names and ages of students who are under 13 years of age
Translated by https://translate.com/machine-translation	
نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۳ سال است	Show the names and ages of students who are under 13 years of age
Translated by Google Translate	
نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۳ سال است	Show the name and age of the students who are less than 3 years old

3.4 Extracting Data from Text

The Afghan language interface to the database used Google speech recognition to extract input sentences from a user speaking the Afghan language. So changing speech to text for performing lexical analysis, syntax analysis, and semantic analysis phases. The extract input sentences are actually the Afghan natural language query in the first step, after a few processes are changed to the corresponding Structured Query Language. Nowadays, in the age of information, like the internet and the web, work and communication ways have changed. There is a lot of information available for the public. We cannot use the available amount of information to process the information and extract knowledge from it, there is not a suitable technique. Text mining is one of the technologies applied for those who deal with unstructured data. To make it easy to use to access and extract information easily and effectively without extra effect, speech is applied. Today for accessing information using just interacts with the system by command and voice, that system recognizes the voice and appeals the necessary module for the output. For converting a speech signal to a word or other language, an automatic speech recognition system uses the process implementing the algorithm as a computer program. Process and access needed information from unstructured data in a short amount of time, called information extraction. The automatic Speech Recognition process contains the Front-end unit, Decoder unit, language models unit shown in figure 3.2 [360].

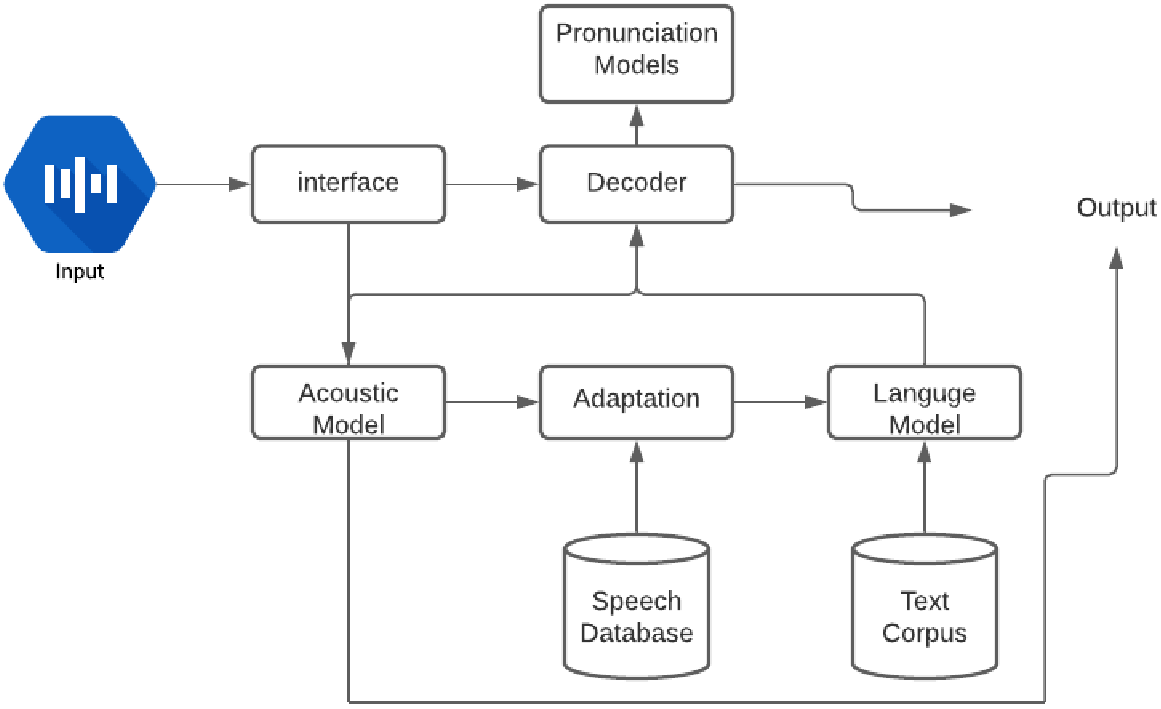


Figure 3.2: Basic model of speech recognition system

Given the figure description, each model and speech database is given below:

- **Acoustic model:** for each different phone, mostly feature victor and context-dependent properties.

- **Phonetic dictionary:** contain a mapping from word to phone. It's not effective because only two or three pronunciation variants are noticed in this, the mapping part is effectively done by some complex function learned by the machine learning algorithm.

- **Language model:** restricting the word search is done with this part. Show different patterns between different words that define which word can follow preceding recognized words and then significantly restrict it. The matching process by neglecting the word not matched. The commonly used language model is the n-gram language model which includes statistics of each word sequence and the finite language model which defines speech sequences with weights. Vocabulary restrictions in the mean issue. It includes chunks like sub-word. Recognition accuracy is lower with a word-based language model.

- **Speech database:** it has an important use in automatic speech recognition. They are also used in various systems like coding, speech synthesis, and analysis which includes language identification and verification process. These applications require large amounts of a database.

3.5 Lexical Analysis

Lexical analysis is a process of decomposing text into words, phrases into meaningful elements. Lexical analysis work based on word level analysis which focuses on the meaning of the word phrases and symbols. The lexical analysis describes the tokenization process. By eliminating white space, comments in the input sentences and converted into series of tokens. The concept of lexical is to take input character sentences and change it to tokens. What preparation of data or text has been required so far? Approximately 90% of the world's data is unstructured and may take the form of text, images, video, or audio. It should take the shape of a web page, HTML document, or other formats, and it should have a lot of noise that has to be addressed and go through a few preparatory steps. Preprocessing involves converting raw text data into a comprehensible format. For instance, real-world data is frequently insufficient, unreliable, and full of noise. Data preparation prepares raw text data for upcoming processing; however, it contains numerous mistakes.

3.5.1 Case Folding

Text normalization is done via case folding. That is, converting lowercase text data to make sure "DATASET" and "dataset" are regarded the same and to have all the data consistent. We are utilizing Lower() function of Python. With this technique, a string of uppercase characters is converted to lowercase ones. This section demonstrates step by step how it functions. Using Python, we convert any text or documents to lowercase. A list of strings is generated and assigned to a variable.

text=['Show the name and Age of the Students who are less than 10 years old']

To execute lower() function on the text data, we show an example that applies case folding to strings directly, as shown below.

```
original_text = text
normalized_text = original_text.lower()
print(normalized_text)


# output
show the name and age of the students who are less than 10 years old.
```

That is how we change the input sentences into lowercase [234].

3.5.2 Eliminating Punctuation Marks

It's a traditional method used in data mining and information retrieval for pre-processing punctuation removal. The removal of punctuation is important at this time. We can boost computing efficiency and shrink the size of the data text corpus. We may eliminate punctuation from the text data by utilizing the regex and replace() routines. As we discussed at the very beginning of this chapter, there are several tools and patterns, including NLTK, spaCy, Gensim, and others. Here, we'll illustrate punctuation removal below.
in this section, we are creating a string to assign to a variable.

```
text=['show the name @ and age of the students who are less than 10 years old.']
import re
s = text
s1 = re.sub( r'[^\w\s]', '',s )
s1 = s1.lower().split()
s1 = " ".join(s1)
s1

#out
'show the name and age of the students who are less than 10 years old'
```

3.5.3 Tokenizing Texts

The fundamental preprocessing step for almost all NLP tasks tokenization in other words, the process of splitting text into smaller units called tokenization as we say, it's a fundamental preprocessing step for almost all NLP applications like sentiment analysis, question answering, natural language processing interface to database, machine translation, information retrieval etc. There are many libraries which are used for tokenization like NLTK, SpaCy, and TextBlob. The simplest way for tokenization is using the TextBlob library. We are following steps for better understanding.
in this section, we are creating a string to assign to a variable.

```
text=['show the name and age of the students  who are less than 10 years old']

#using NLTK
import nltk
#create data
mystring = text
nltk.word_tokenize(mystring)
```

#out ['show', 'the', 'name', 'and', 'age', 'of', 'the', 'students', 'who', 'are', 'less', 'than', '10', 'years', 'old']

```
#using split function from python
mystring.split()
```

#out ['show', 'the', 'name', 'and', 'age', 'of', 'the', 'students', 'who', 'are', 'less', 'than', '10', 'years', 'old']

3.5.4 Removing Stop Words

Stop words are popular terms that have less meaning than other keywords, since they have no inherent meaning. We must eliminate less often used terms to concentrate on vital keywords. If you search, the query "How do I use Google." using the python search engine tried to find web pages that contained the terms "how", "do", "I", "use", and "Google,". The search engine finds numerous pages that include the words "how", "do" and "I" then the pages contain information about Google the term "how", "do", and "I" are commonly used in the English language. If you remove such terms, this search engine will focus on retrieving information that includes the keyword "Google" to retrieve the actual page that we are interested in.

By using a simple NLTK library, we are showing how to remove stop words from a carpus or documents.

```
text= "show the name and age of the students  who are less than 10 years old"
#!pip install nltk
import nltk
#nltk.download()
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
mystring = text
all_words=word_tokenize(mystring)
#remove stop words
stop = stopwords.words('english')
filtered_words = [word for word in all_words if word not in stop]
print(filtered_words)
#out
['show', 'name', 'age', 'students', 'less', '10', 'years', 'old']
```

3.6 Syntax Analysis

Syntactic analysis plays a key role in understanding grammatical relationships between words. It helps to know the meaning of the sentences. The way to decompose or parse sentences into their constituents, group of words, or words that functions as a single role. The constituency parse tree means to decompose the hierarchy into small units until each unit consists of a word. For example, "Ahmad teaches a course" can be divided down into a noun phrase that includes “Ahmad” and a noun phrase that has “a course”, the phrase a course can be divided down by the verb “teaches” and a noun phrase that has “a course” the expression “a course” broken to the determiner “a” and then noun “course” This parse is shown in the figure below. [202].

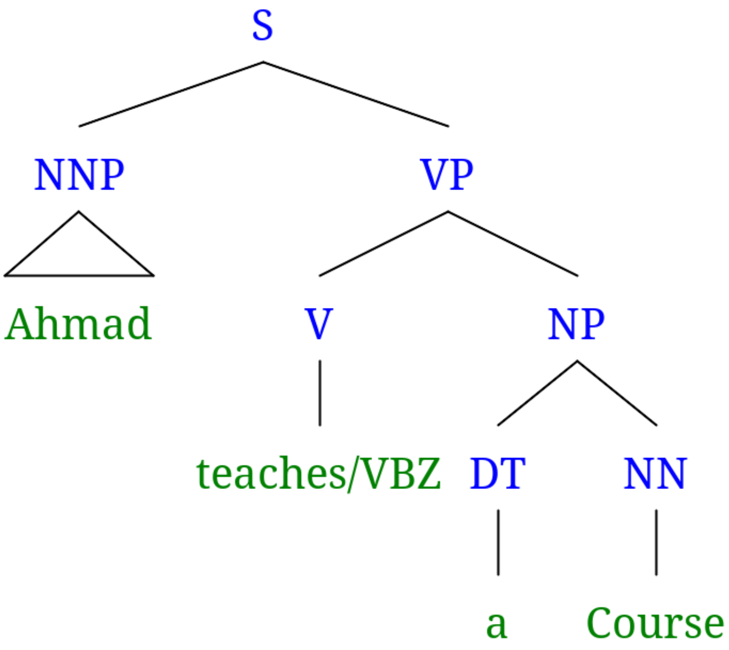
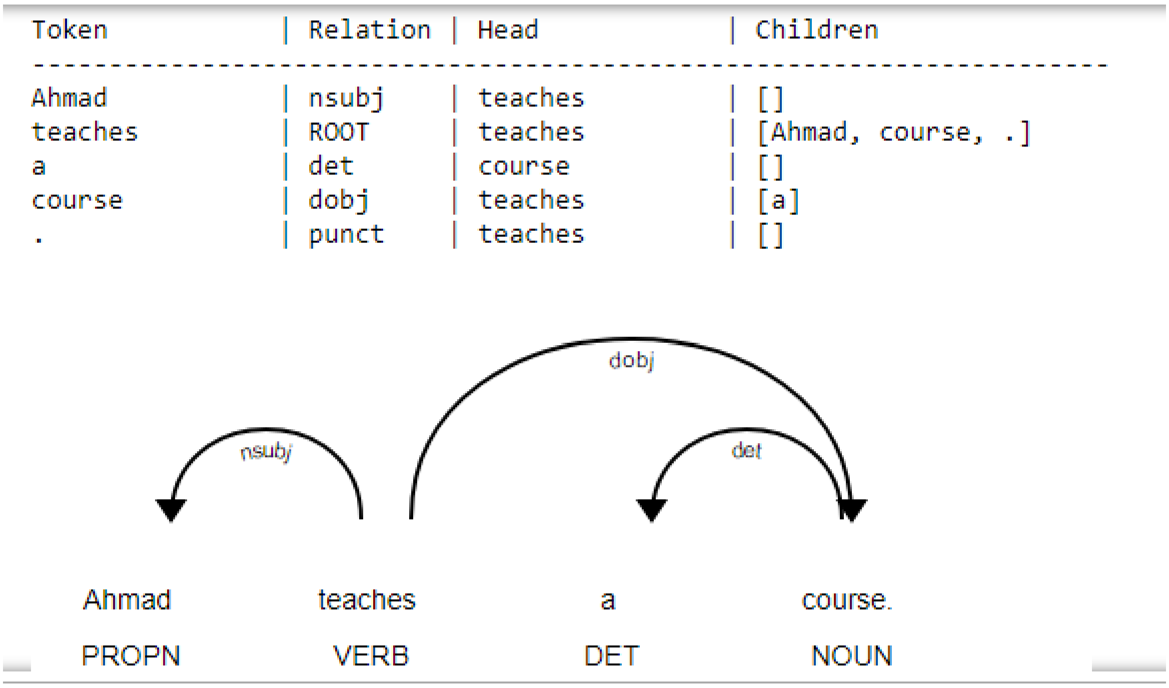


Figure 3.3: Syntactic parse of the sentence “Ahmad teaches a course”

A dependency parser, show the relationship between two-word pair in a sentence each relationship as head-to-dependent or dependent-to-head, each dependency is assigned a grammatical relation type, based on the relation type that the pattern is given for.



```
Token              | Relation | Head               | Children
-------------------------------------------------------------------
Ahmad              | nsubj    | teaches            | []
teaches            | ROOT     | teaches            | [Ahmad, course, .]
a                  | det      | course             | []
course             | dobj     | teaches            | [a]
.                  | punct    | teaches            | []
```

Figure 3.4: Dependency parse of the sentence “Ahmad teaches a course”

3.6.1 Stemming/Lemmatization.

“runs”, “running”, and “ran” are represented in various forms due to morphological modifications in this example, derivationally related with similar meanings such as “change” and “changes”, and “changing” Stemming and Lemmatization are two common techniques to relate such words. Stemming is the process of taking away words’ derivational affixes from their ends to their stems. Utilizing “dictionary and morphological analysis of words” to eliminate inflectional ending only and to return the lemma is a lemmatization understood as the lexicon form of a word. Due to the above example, the meaning of the word is similar, but it’s hard for a machine to understand the actual root of the word. For humans, it is easy that way to remove the suffix of the word. It is easy for a machine to process the word that is 100 times mentioned in the corpus to know the meaning of an accurate word. To avoid confusion. In the example given below, we are showing the stemming technique using python.

```
text= "show the name and age of the students  who are less than 10 years old"

#!pip install nltk

import nltk

#nltk.download()

from nltk.corpus import stopwords

from nltk.tokenize import word_tokenize
```

```
mystring = text
all_words=word_tokenize(mystring)
#remove stop words
stop = stopwords.words('english')
filtered_words = [word for word in all_words if word not in stop]
#Code for lemmatize
from nltk.stem import PorterStemmer
ps =PorterStemmer()
for w in filtered_words:
    rootWord=ps.stem(w)
    print(rootWord)
#out
[show name age student less 10 year old]
```

#You can observe that students and years are lemmatized to student and year, as explained.

3.7 Semantic Analysis

For knowing the user's emotion and deciding what is expressed through them. The aim of semantic analysis is to know the meaning of language. Semantics agreements with the meaning of sentences and words and are used to evaluate and represent human language and analysis texts written in English or other languages.[269].

3.7.1 Noun-Pronoun-Verb Tagger

The part of speech tagging is used to assign each word in the Corpus to a word category such as noun, verb, adjective. Depending on the context, a word takes more than one part of speech. Word tag POS is used when word categories between very different languages and different tags of even some languages are different, for the English part of speech taggers who use the Penn Tree bank tag set, which has 45 categories. Includes noun, verb, adjective, adverb, preposition ("under", "over"). The major four word categories are noun, verb, adjective, and adverb, which are considered open class. Most sets achieve over 90% accuracy, that means many words are unambiguous, For computing the probability of tag sequences, assigning tags of the word based on hidden Markov model Before and after the target words they are showing an accuracy of close to 97%. After tokenizing the text after the preprocessing phase every word are tokens tagged as noun, pronoun, or verb the various tags with their description or declared below.

Table 3.4: POS tags and descriptions

Tags	Explanation	Instance
CC	Conjunction	And, or, but
CD	Cardinal Number	Five
NN	Noun	Softness, cat
NNs	Nouns	An animal, a quality
PROPN	Proper noun: First name, people, towns, etc.	Sadullah, Afghan, Kabul
INTJ	Interjection: To show astonishment, goodness, Gosh, etc.	Ouch, Hey, well, oh no
RB	Adverb	Extremely, hard
AUX	Auxiliary: assisting verb tagging tense, modality, feeling, etc.,	Have, do, would, being
DET	Determiner: indicate noun expression properties	My, each, some, any
ADJ	Adjective: noun moderator representing possessions	Beautiful, jealous, she is happy
PUNCT	Punctuation	˙, , ()
SYM	Symbols like $ or emoji	˙$, %

Output: [('cities', 'NNS'), ('located', 'VBN'), ('Afghanistan', 'NN')]

3.8 Ambiguity Remover

Various English words have multiple meanings, and many relatively common terms have numerous meanings in the language. The table 3.5 depict the most number of feelings of a word in the Merriam-Webster pocket dictionary (“information from Merriam-Webster website, 2022”).

Table 3.5: Different senses of a word in English language

Word	Category	No. of senses	Word	category	no. of senses
go	Verb	over227	take	Verb	over 120
fall	Verb	over 91	dead	Adj	over 145
run	Verb	over 177	good	Adj	over 275
turn	Verb	over 77	have	Verb	over 73
way	Noun	over 92	line	Nouns	over 114
work	Verb	over 86	pass	Verbs	over 111
do	Verb	over 220	touch	Verbs	over 33
draw	Verb	over 52	dry	Adj	over 57
play	Verb	over 52	wing	Nouns	over 20
get	Verb	over 269	draft	Nouns	over 25
form	Noun	over 72	give	Verb	over 90
make	Verb	over 171	turns	Noun	over 63
strike	Verb	over 103	huge	Adj	over 88

the number of useful senses mentioned in the Merriam-Webster pocket dictionary for practical reasons, natural language understanding systems must be able to disambiguate words having different meanings, and knowledge representation and the semantic interpretation techniques utilized to do so must be compatible with the system. HOMONYMY, CAT-EGORIAL AMBIGUITY, and POLYSEMY are the three categories of lexical ambiguity. Polysemous words have meanings that are connected to one another. The open verb, for example, can refer to expanding, unfolding, and revealing, shifting to an open position, producing openings in, and so on. Homonymic words, on the other hand, have meanings that have no link to one another; for example, bark refers to both the sound the dog makes and the things on the outside of a tree. A word can be both polysemous and homonymous; the adjective right has several meanings in terms of accuracy and fairness. But also senses concerning the right-hand side. There is no clear distinction between homonymy, polysemy, and metaphor. Today's metaphor is perhaps tomorrow's homonymy or polysemy. For example, there is an obvious association between the mouth in the person's mouth sense and the mouth of a river sense. But for applicable objectives, are quite dissimilar ideas, and it is not clear where the mouth should be positioned. These terms are categorically ambiguous and are syntactic categories that can alter. For instance, the word "sink" can refer to a plumbing device or a verb that means to immerse. Categorical ambiguity differs from the other kind in that respect ambiguity is categorical and polysemous because its noun and verb meaning are connected, whereas sink's ambiguity is categorical and homonym since its noun and verb meaning is unconnected. Categorical ambiguity is mostly a parsing issue. Thus, I won't go into further detail about it in this thesis, except for how it interacts with other sorts of ambiguity "replicate Milne 1980, 1986 for discussion of handling categorical ambiguity in a deterministic parser". Nouns must be homonymous, whereas verbs must be polysemous. Of course, there are numerous homonymous verbs and polysemous nouns, which is consistent with Gentner's (1981a, 1981b) claim that verbs are more adjustable than nouns. Nouns typically refer to fixed entities, whereas verb meanings are easily adjusted to fit the context, with frequent adjustments becoming lexicalized as new but real. Although research, including his own, has demonstrated that people's institutions do differentiate between polysemy and homonymy, Panman (1982) contends. Although it seems strange that a cognitively real linguistic phenomenon should have no place in linguistics theory, I will also make little use of it in this work. It will typically, but not always, be a matter of indifference how closely related those two entities are. Maintaining the distinction at the level of linguistic theory is difficult and probably unnecessary. We can distinguish between words using a system that understands natural language, but it has to consider both the discourse context in which the word appears, and local clues found inside the sentences themselves. [5].

3.8.1 Relations-Attributes-Clauses Identifiers

Organizations are not widely used the NLIDB systems because they cannot process the ambiguity and complexity of natural language, which provides them with more errors and put them at risk, That's why we need to capture even the smallest information of the natural language query before converting into SQL Query. The relational database includes tables for storing information, the tables include columns and rows every column in the table is called an attribute; its query consists of different SQL clauses like SELECT, FROM, WHERE, GROUP BY, HAVING, and ORDER BY, So every clause in SQL Query have attributed information, the attribute is significant and effective for translating natural language query to SQL Query, The attribute writing by the user as a natural language query text is called an explicit attribute, final a natural language query translate to SQL Query, explicit attribute can belong to different Structured Query Language clauses. There are two types of attributes that can be found in natural language queries.

Explicit Attribute: It's directly mentioned by the user in the natural language query.

Implicit attributes: It is not directly mentioned by the user in a natural language query. This attribute is identified thanks to values mentioned in the natural language query by the user. Domain dictionaries are used for identifying this attribute. We use a system dictionary that contains Persian words with corresponding English words, and the users add NLQ queries with the help of the system dictionary to know when the user wants to insert or delete records [137] [355].

Table 3.6: System lexicon

English Word	Corresponding Persian Word
Student	شاگرد، متعلم
From	از
Name	نام
Select	انتخاب کردن، نشان دادن
Where	که، آن
Roll_no	شماره سریال
Delete	حذف کردن ، برداشتن

3.8.2 Query Generation

Conversation of the user query, It needs to set off rules and behavior Convent user query statement into SQL Query and then retrieve data from database use semantics matching procedure to change natural language question to relative SQL Query, we described various step in the early stage of this chapter like lowercase conversion, tokenization, ambiguity remover that we are using for translate which is mapped to the database to retrieve required information, we are describing the SQL Query conversion in the example below [178].

1. input چند دانش آموز در کلاس عدد ۵ وجود دارد

2. tokenize input to:

دارد	وجود	۵	عدد	کلاس	در	آموز	دانش	چند

3. Persian input words translate into English words

there	are	several	students	in	class	number	5

4. Tagger tags the words: [(‘there’, ‘EX’), (‘are’, ‘VBP’), (‘several’, ‘JJ’), (‘student’, ‘NN’), (‘in’, ‘IN’), (‘class’, ‘NN’), (‘number’, ‘NN’), (‘5’, ‘CD’)]

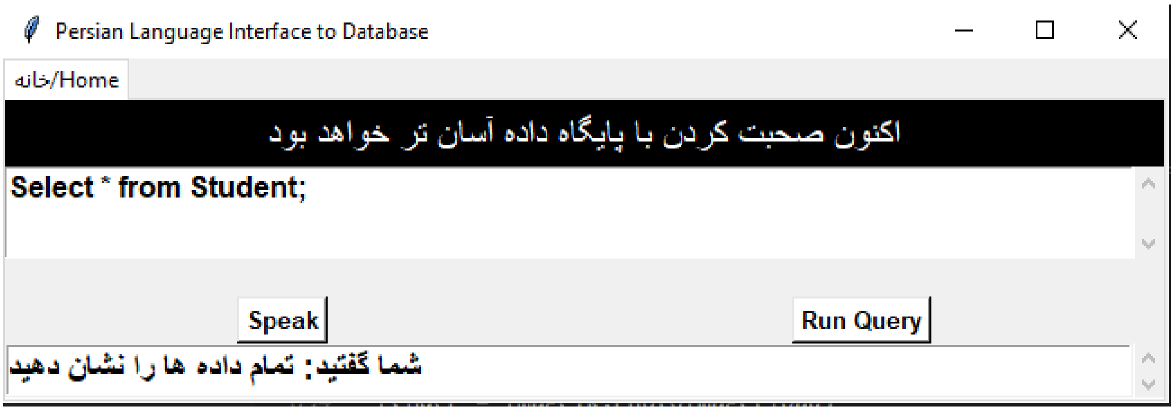
5. Through syntactic parsing, tokens maps to table name, condition and operation, columns name.

6. Finally, the query is generated:
   Select COUNT(*) from Student where class = ’5’;

Chapter 4

Result

Afghan Spoken natural language interfaces to the database work when the user puts a spoken natural language query. In the Afghan language in the form of audio, we used Google speech recognition, Google Speech Recognition accepts more than 90 languages in the world. Google speech recognition is reasonable for voice detection special in the Persian language. After the system receives input from the user, in the second step, Google Translate is used for translating the Afghan language into the corresponding English language. Currently, we created a database for students with basic attributes or identifiers. When a Persian text translates into English, then, data preprocessing applies to translate sentences like tokenization, part of speech tagging, etc. For analysis, we test Persian sentences and English sentences type could be selection, insertion, count, conditional selection, and removing records, which are identified by Input sentences or words. The tokens are mapped with database values. After mapping, the system generates a SQL query and show it in the graphical user interface to the user [355].

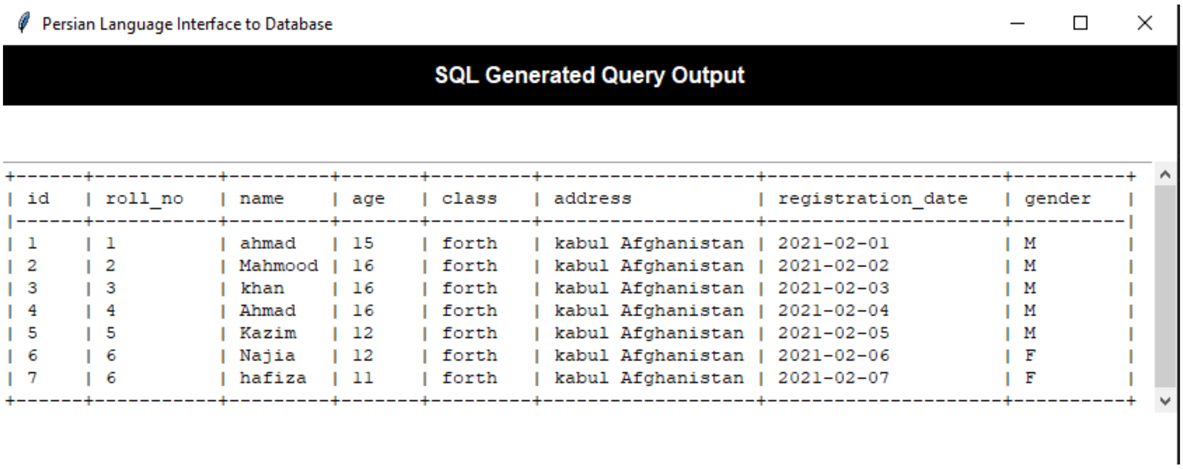


اکنون صحبت کردن با پایگاه داده آسان تر خواهد بود

Select * from Student;

شما گفتید: تمام داده ها را نشان دهید

Figure 4.1: Select data from MySQL database



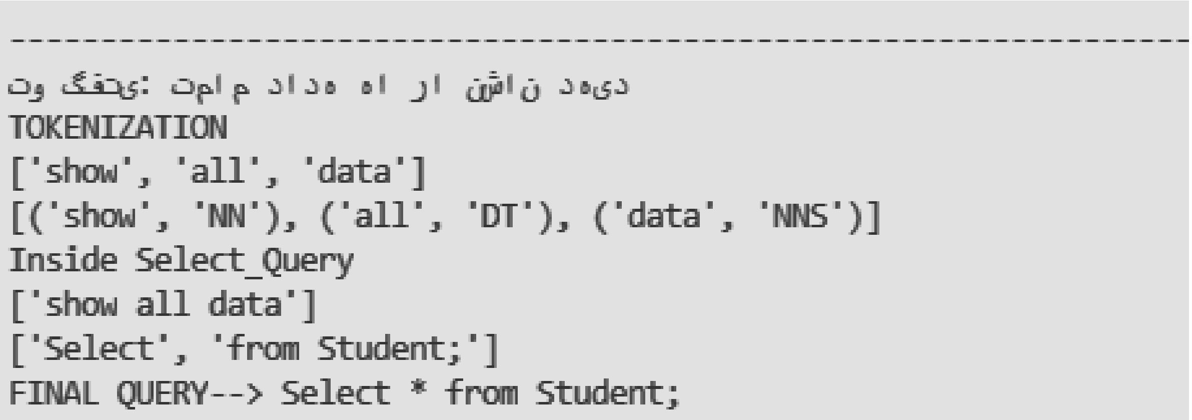
SQL Generated Query Output

```
+------+-----------+---------+-------+---------+-------------------+---------------------+----------+
| id   | roll_no   | name    | age   | class   | address           | registration_date   | gender   |
|------+-----------+---------+-------+---------+-------------------+---------------------+----------|
| 1    | 1         | ahmad   | 15    | forth   | kabul Afghanistan | 2021-02-01          | M        |
| 2    | 2         | Mahmood | 16    | forth   | kabul Afghanistan | 2021-02-02          | M        |
| 3    | 3         | khan    | 16    | forth   | kabul Afghanistan | 2021-02-03          | M        |
| 4    | 4         | Ahmad   | 16    | forth   | kabul Afghanistan | 2021-02-04          | M        |
| 5    | 5         | Kazim   | 12    | forth   | kabul Afghanistan | 2021-02-05          | M        |
| 6    | 6         | Najia   | 12    | forth   | kabul Afghanistan | 2021-02-06          | F        |
| 7    | 6         | hafiza  | 11    | forth   | kabul Afghanistan | 2021-02-07          | F        |
+------+-----------+---------+-------+---------+-------------------+---------------------+----------+
```

Figure 4.2: SQL query generated result for SELECT clause

4.1 Pre-Processing Data

Our system, after taking input in the Persian language and translating it to corresponding English sentences, the system tokenizes the translated sentences using the Python NLTK library. They are many types of tokens in the sentences, they have been recognized using the port of speech tagging, the tokens include adverb, adjective, cardinal number, conjunction, determiner, foreign word, infinitive, interjection, list marker, model, noun, particle, pronoun, symbol, verb, etc., which is illustrated in figure 4.3. Then after identifying the POS tags, the token mapping to the corresponding attribute clauses like SELECT, WHERE, etc., after mapping the final SQL query is generated for retrieving data from the database [355].

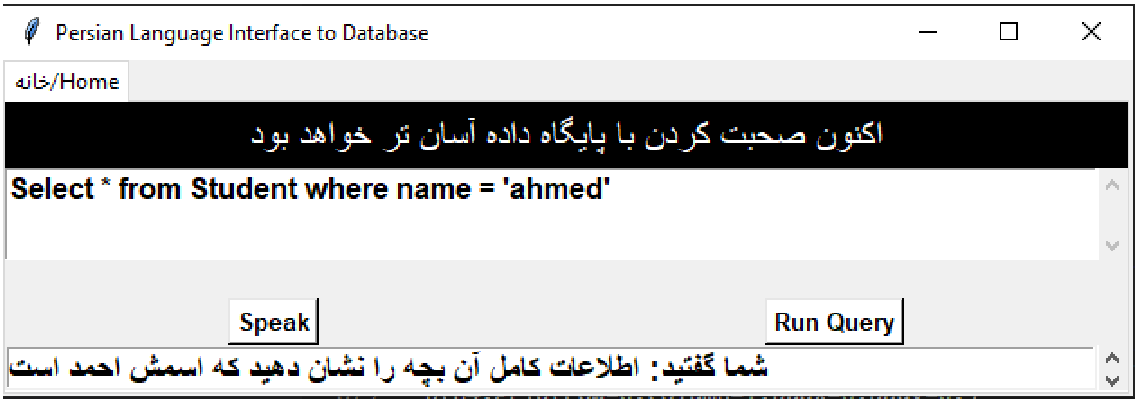


```
--------------------------------------------------------------------
وت یحقگ: تامم هداد اه ار ناشن دیهد
TOKENIZATION
['show', 'all', 'data']
[('show', 'NN'), ('all', 'DT'), ('data', 'NNS')]
Inside Select_Query
['show all data']
['Select', 'from Student;']
FINAL QUERY--> Select * from Student;
```

Figure 4.3: Pre-processing data

4.2 Natural Language Query for "WHERE" Clause

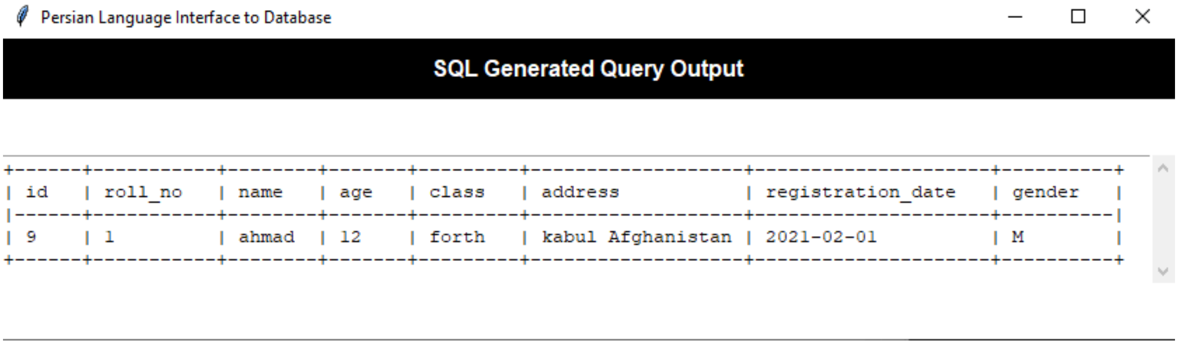
Retrieving data from the database needs knowledge of Structured Query Language. Accessing data needs the knowledge of domain expert users, which is a challenge for inexpert users all around the world. The Spoken Afghan natural language control interface to the database is built to come up with current challenges. And built an interface for a user that can access the database in their native language. We built the current system for Persian native speakers around the words. Who don't know how to write the specific natural language query with their correct spelling, this application is built for those who have disabilities? It's easy to speak with the system and retrieve specific information from a given database, in figures 4.4 and 4.5, illustrated how the system received input and processed it with the corresponding "FROM" and "WHERE" clauses of SQL [355].


Persian Language Interface to Database
خانه/Home
اکنون صحبت کردن با پایگاه داده آسان تر خواهد بود
Speak
Run Query

Select * from Student where name = 'ahmed'

شما گفتید: اطلاعات کامل آن بچه را نشان دهید که اسمش احمد است

Figure 4.4: Query with "WHERE" clause

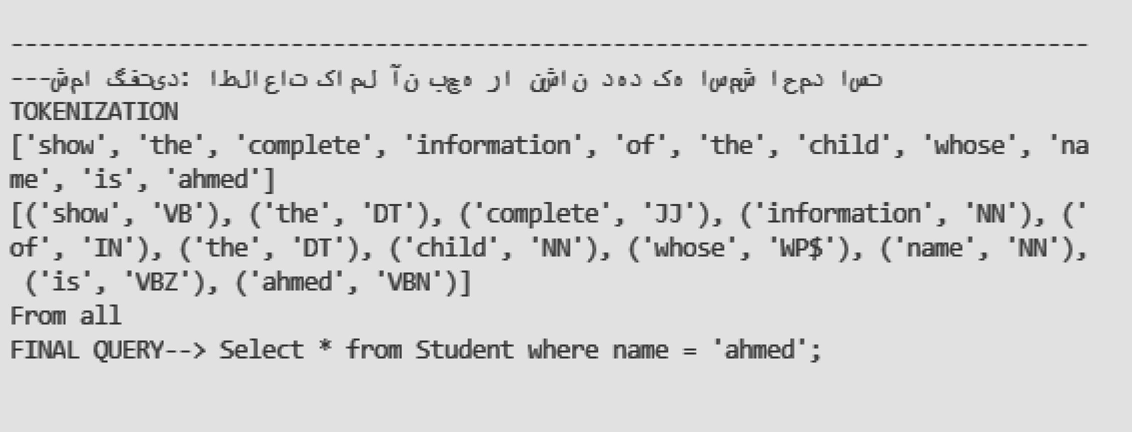

Persian Language Interface to Database
SQL Generated Query Output

```
+------+-----------+--------+-------+---------+-------------------+----------------------+----------+
| id   | roll_no   | name   | age   | class   | address           | registration_date    | gender   |
|------+-----------+--------+-------+---------+-------------------+----------------------+----------|
| 9    | 1         | ahmad  | 12    | forth   | kabul Afghanistan | 2021-02-01           | M        |
+------+-----------+--------+-------+---------+-------------------+----------------------+----------+
```

Figure 4.5: SQL query generated result for "WHERE" clause

4.3 Query Generation Process for "WHERE" Clause

The process for selecting specific records from the database using "FROM" and "WHERE" clauses are easy for domain expert users. To use spoken natural language query for retrieving specific data from the database using the Afghan language. In our application, when the user inputs a query in a natural language using the Persian language system, it is translated to corresponding English sentences. Then tokenize the sentences into individual words. After using part of speech, tagging each word with their relative grammar like a verb, noun, pronoun, adjective, adverb, preposition, etc. Then every word's map to their corresponding attribute name or clause for generating SQL Query in figure 4.6, which is illustrated clearly [355].

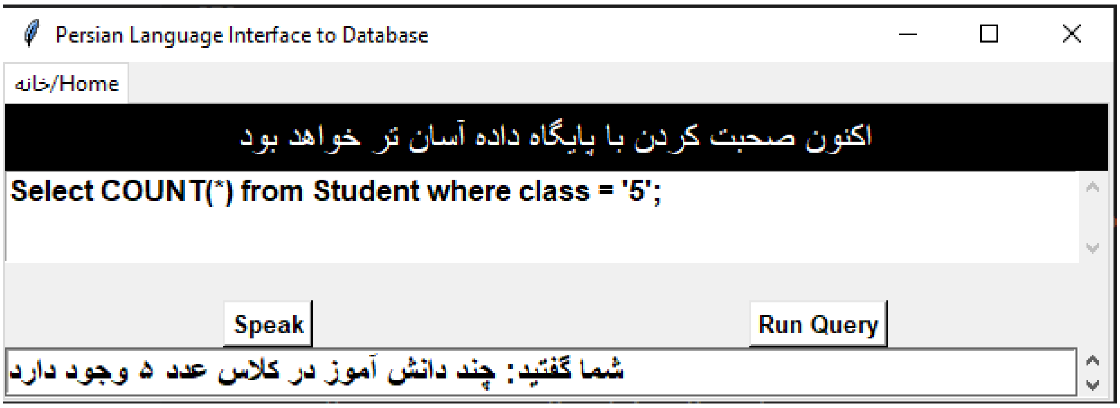


```
-------------------------------------------------------------------------------
---شما گفتید: اطلاعات کامل آن بچه را نشان دهد که اسمش احمد است
TOKENIZATION
['show', 'the', 'complete', 'information', 'of', 'the', 'child', 'whose', 'na
me', 'is', 'ahmed']
[('show', 'VB'), ('the', 'DT'), ('complete', 'JJ'), ('information', 'NN'), ('
of', 'IN'), ('the', 'DT'), ('child', 'NN'), ('whose', 'WP$'), ('name', 'NN'),
 ('is', 'VBZ'), ('ahmed', 'VBN')]
From all
FINAL QUERY--> Select * from Student where name = 'ahmed';
```

Figure 4.6: Query Generation (step by step)

4.4 Using NL Query for COUNT() Function

The application of the Persian language interface to the database supports many SQL clauses like SELECT, WHERE, FROM, and one of the important structure query clauses is COUNT () which is used as a return function for the number of records returned by a select query. This function will not be used to count null values. By using the function, it's easy for Persian speakers to count records like how many students or in the class or for a business like how many products are available in stock in many domains. It helps organizations and makes work easy for inexpert employees who didn't know how to use Structured Query Language, as figures 4.7, 4.8 and 4.9 are illustrated clearly for better understanding [355].

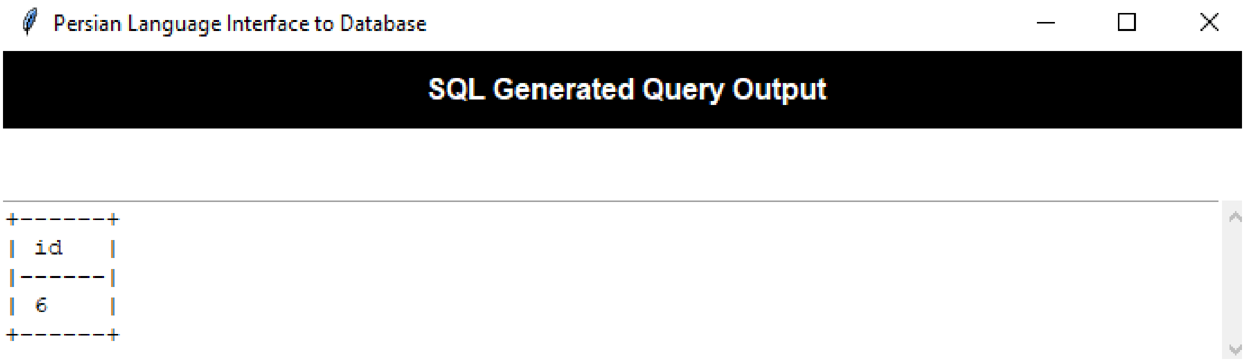


اکنون صحبت کردن با پایگاه داده آسان تر خواهد بود

Select COUNT(*) from Student where class = '5';

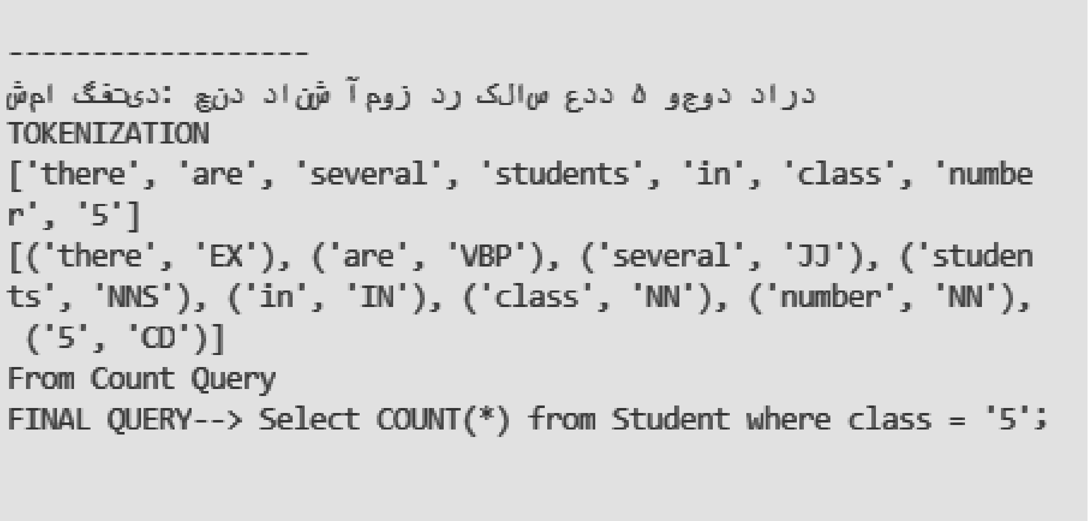
شما گفتید: چند دانش آموز در کلاس عدد ۵ وجود دارد

Figure 4.7: COUNT function query


Persian Language Interface to Database
SQL Generated Query Output

```
+------+
| id   |
|------|
| 6    |
+------+
```

Figure 4.8: SQL query generated for COUNT function

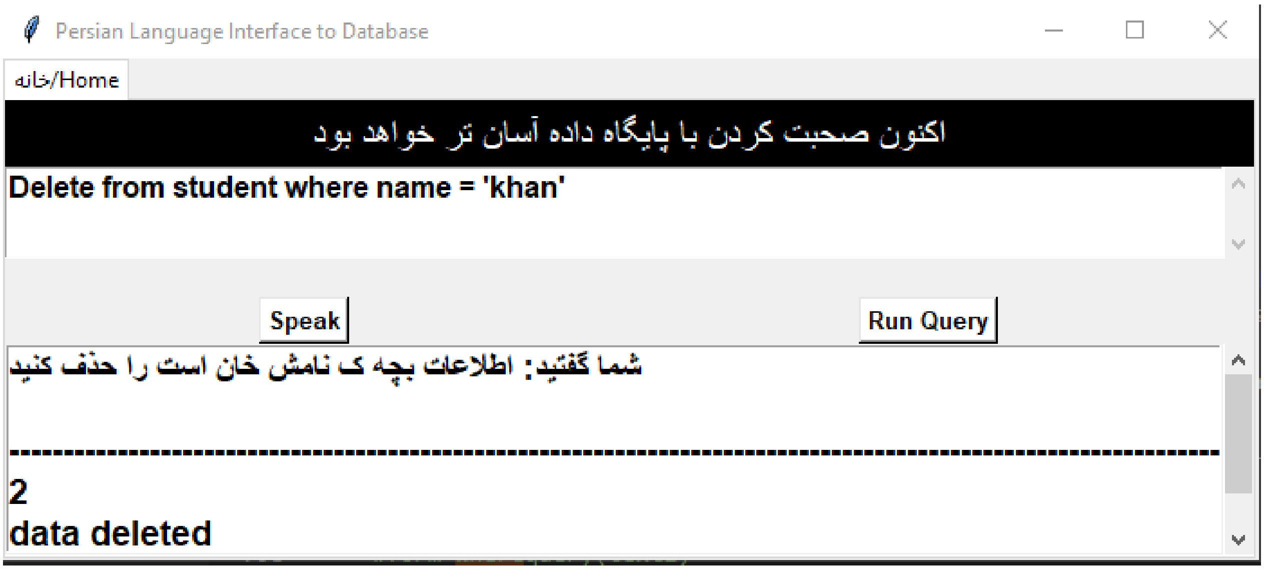


```
-------------------
شما گفتید: چند دانش آموز در کلاس عدد ۵ وجود دارد
TOKENIZATION
['there', 'are', 'several', 'students', 'in', 'class', 'numbe
r', '5']
[('there', 'EX'), ('are', 'VBP'), ('several', 'JJ'), ('studen
ts', 'NNS'), ('in', 'IN'), ('class', 'NN'), ('number', 'NN'),
 ('5', 'CD')]
From Count Query
FINAL QUERY--> Select COUNT(*) from Student where class = '5';
```

Figure 4.9: COUNT function query generation process

4.5 Using NL Query for DELETE Clause

By using a delete query in the Afghan spoken natural language interface to the database using the Persian language, our application supports “SELECT”, “WHERE”, “FROM”, and “COUNT ()” functions. It also supports a DELETE clause to delete either table or specific record from the database. Our application will receive input in the Persian language from the user, then after a process which is illustrated in figures 4.10 and 4.11 clearly for better understanding [355].

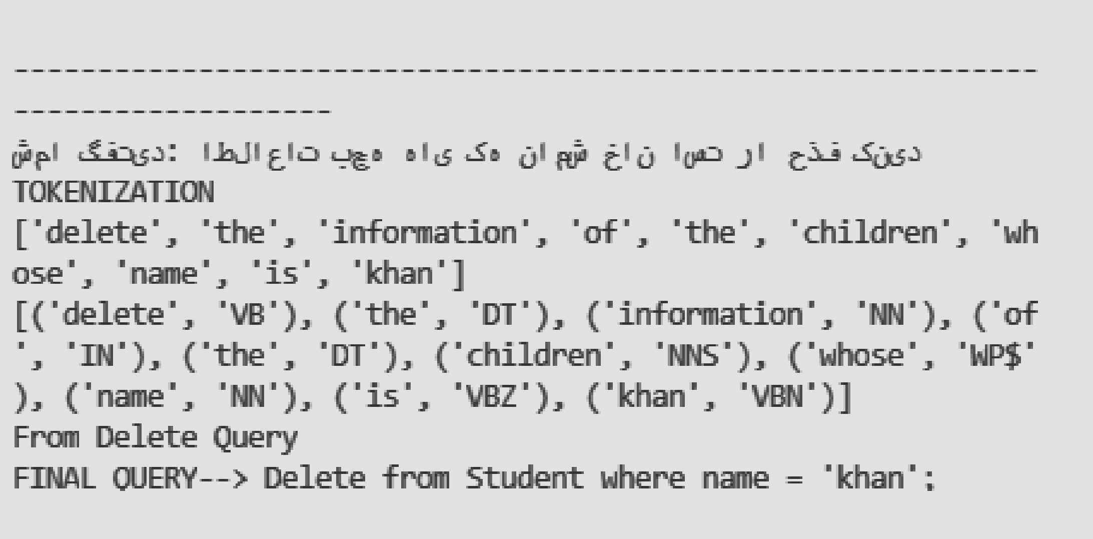

Persian Language Interface to Database
خانه/Home
اکنون صحبت کردن با پایگاه داده آسان تر خواهد بود
Speak
Run Query

```
Delete from student where name = 'khan'
```

```
شما گفتید: اطلاعات بچه ک نامش خان است را حذف کنید

-----------------------------------------------------------------------------------------------------------
2
data deleted
```

Figure 4.10: DELETE clause query



```
---------------------------------------------------------------
-------------------
امش دیتفگ: تاعالطا هچب یاه هک شمان ناخ تسا ار فذح دینک
TOKENIZATION
['delete', 'the', 'information', 'of', 'the', 'children', 'wh
ose', 'name', 'is', 'khan']
[('delete', 'VB'), ('the', 'DT'), ('information', 'NN'), ('of
', 'IN'), ('the', 'DT'), ('children', 'NNS'), ('whose', 'WP$'
), ('name', 'NN'), ('is', 'VBZ'), ('khan', 'VBN')]
From Delete Query
FINAL QUERY--> Delete from Student where name = 'khan';
```

Figure 4.11: DELETE clause query generation

4.6 Result Evaluation

For testing our Afghan spoken natural language interface to the database system, we have utilized a database of students. A bunch of 50 spoken natural language queries is used. For efficiency measurement, we have employed a confusion matrix consisting of True Positive (TP). False Positive (FP), True Negative (TN), and False Negative (FN) [355].

Table 4.1: Confusion Matrix For 50 Persian NLQ

	True Predicted	False Predicted
Actual True	TP=41	FN=4
Actual False	FP=1	TN=4

Thanks to the formula, we have measured Accuracy, Precision, Recall, and a F_1 score.

$$Precision, P = \frac{TP}{TP + FP} \tag{4.1}$$

$$Recall, R = \frac{TP}{TP + TN} \tag{4.2}$$

$$Accuracy, A = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.3}$$

$$F_1 = \frac{2PR}{P + R} = \frac{2TP}{2TP + FP + FN} \tag{4.4}$$

Precision	Recall	Accuracy	F_1 Score
97	91	90	94

50 Easy Persian spoken questions with various versions were assessed. The system now functions using the straightforward aggregate operation. Future system upgrades are required for complicated inquiries like “GROUP BY,” “HAVING,” and nested queries.

Table 4.2: Afghan Natural Language Query list

No	Persian Sentences	Translated to English	SQL Query
1	تمام داده ها را نشان دهید	Show all data	`Select * from Student;`
2	تمام اطلاعات را نشان دهید	Show all information	`Select * from Student;`
3	نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۳ سال است	Show the name and age of the students who are less than 13 years old	`Select name, age from Student where age < 13;`
4	نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۲ سال است	Show the name and age of the students who are less than 12 years old	`Select name, age from Student where age < 12;`
5	چند دانش آموز در کلاس عدد ۴ وجود دارد.	There are several students in class number 4	`Select COUNT(*) from Student where class = 4;`
6	چند دانش آموز در کلاس عدد ۵ وجود دارد.	There are several student in class number 5	`Select COUNT(*) from Student where class = 5;`

No	Persian Sentences	Translated to English	SQL Query
7	چند دانش آموز در کلاس عدد ۱۰ وجود دارد.	There are several student in class number 10	Select COUNT(*) from Student where class = 10;
8	چند دانش آموز در کلاس عدد ۹ وجود دارد.	There are several student in class number 9	Select COUNT(*) from Student where class = 9;
9	سن چند دانش آموز ۱۲ سال است.	The age of several students is 12 years.	Select COUNT(*) from Student where age =12;
10	سن چند دانش آموز ۱۳سال است.	The age of several students is 13 years.	Select COUNT(*) from Student where age =13;
11	سن چند دانش آموز بزرگتر از ۱۲ سال است.	The age of several students is older than 12 years.	Select COUNT(*) from Student where age > 12;
12	سن چند دانش آموز بزرگتر از ۱۳ سال است.	The age of several students is older than 13 years.	Select COUNT(*) from Student where age > 13;
13	سن چند دانش آموز کوچکتر از ۱۲ سال است.	The age of several students is younger than 12 years.	Select COUNT(*) from Student where age < 12;
14	سن چند دانش آموز کوچکتر از ۱۳ سال است.	The age of several students is younger than 13 years.	Select COUNT(*) from Student where age < 13;
15	شهر چند دانش آموز کابل است.	The city of several students is Kabul	Select COUNT(*) from Student where city = 'Kabul';
16	شهر چند دانش آموز مزارشریف است.	The city of several students is Mazar-i-Sharif	Select COUNT(*) from Student where city = 'Mazar-i-Sharif';
17	شهر چند دانش آموز خوست است.	The city of several students is khost	Select COUNT(*) from Student where city = 'khost';
18	شهر چند دانش آموز کندز است.	The city of several students is kunduz	Select COUNT(*) from Student where city ='kunduz';
19	نام و سن دانش آموزان را نشان دهید	Show students names and age	Select name, age from Student;
20	نام و شهر دانش آموزان را نشان دهید	Show students names and city	Select name, city from Student;
21	سن و شهر دانش آموزان را نشان دهید	Show students age and city	Select age, city from Student;
22	اطلاعات کامل بچه را نشان دهید که اسمش احمد است	Show the full information of the student whose name is ahmad.	Select * from Student where name= 'ahmad';
23	اطلاعات کامل بچه را نشان دهید که اسمش فضل است	Show the full information of the student whose name is fazal.	Select * from Student where name= 'fazal';

No	Persian Sentences	Translated to English	SQL Query
24	اطلاعات کامل بچه را نشان دهید که اسمش خان است	Show the full information of the student whose name is khan.	Select * from Student where name= 'khan';
25	اطلاعات کامل بچه که نامش خان است را حذف کنید.	Delete the information of the Student whose name is khan	Delete from student where name ='khan';
26	اطلاعات کامل بچه که نامش احمد است را حذف کنید.	Delete the information of the Student whose name is ahmad	Delete from Student where name ='ahmad';
27	نام دانش آموزان را نشان دهید	Show the student names	Select name from Student;
28	آدرس دانش آموزان را نشان دهید	Show students address	Select address from Student;
29	نام سن و کلاس دانش آموزان را نشان دهید.	Show Students age and class, name	Select age, class, name from Student;
30	شاگرد را نشان دهید که اسمش احمد است.	Show student whose name is ahmad	Select * from Student where name = 'ahmad';
31	شاگرد را نشان دهید که اسمش مریم است.	Show student whose name is Maryam	Select * from Student where name = 'Maryam';
32	شاگرد را انتخاب کنید که شماره سریال اش دوم است.	Select student whose serial number is 11	Select * from Student where roll_no = 11;
33	شاگرد را انتخاب کنید که شماره سریال اش چهارم است.	Select student whose serial number is 4	Select * from Student where roll_no = 4;
34	شاگرد را انتخاب کنید که شماره سریال اش ششم است.	Select student whose serial number is 4	Select * from Student where roll_no = 6;
35	شاگرد را انتخاب کنید که شماره سریال اش دهم است.	Select student whose serial number is 10	Select * from Student where roll_no = 10;
36	شاگرد را انتخاب کنید که شماره سریال اش سوم است.	Select student whose serial number is 3	Select * from Student where roll_no = 3;
37	نام وکلاس دانش آموزان را نشان دهید	Show students names and class	Select name, class from Student;
38	کلاس و سن دانش آموزان را نشان دهید	Show students class and age	Select class, age from Student;
39	اطلاعات کامل بچه که نامش محمد است را حذف کنید.	Delete the information of the Student whose name is Mohammad	Delete from Student where name ='Mohammad';
40	نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۸ سال است	Show the name and age of the students who are less than 8 years old	Select name, age from Student where age < 8;

No	Persian Sentences	Translated to English	SQL Query
41	نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۶ سال است	Show the name and age of the students who are less than 6 years old	`Select name, age from Student where age < 6;`
42	نام و سن دانش آموزان را نشان دهید که سن آن کمتر از ۱۵ سال است	Show the name and age of the students who are less than 15 years old	`Select name, age from Student where age < 15;`
43	کلاس دانش آموزان را نشان دهید	Show the student class	`Select name from Student;`
44	شهر دانش آموزان را نشان دهید	Show the student city	`Select city from Student;`
45	تمام داده ها را نشان دهید	Show all Students	`Select * from Student;`

4.7 Results from Existing NLIDB Research

In the area of natural language interface to database, many works done especially in the English language with the high accuracy like M.Llopis et al. [96] design AskMe system which achieve 94% accuracy for changing natural language query into structure query language and A. Giordani et al. [72] designed question translation with generative parser which utilized grammatical dependencies and metadata matching model, which are mentioned in the table 4.3 below. Much research was done in non-English language as well, for instance, H.Bais et al. [144] design GANLIDB system which work based on machine learning and natural language processing. D.T.Nguyen et al. [80] designed Vietnamese NLIDB, which work based on GATE framework and R language. Similar work is done for many languages, but so far, there are no research done until now for the Persian language. Our proposed system achieved more than 90% accuracy in translating Persian query into Structured Query Language. As the datasets and the languages differ, it may not be appropriate to compare the performance among the systems. However, the table is included below for reference only.

Table 4.3: Existing research on Natural Language Interface to Database (NLIDB) for different languages

No	Author	NLIDB System Name	Method	Accuracy (for reference)
1	B. Xu, et al. [252]	NADAQ System	Convolution neural network, Sequence-to-Sequence	83.9%
2	N.Weir et al. [293]	DBpal System	Deep learning model, Seq-seq recurrent neural network	75.93%
3	F.Li et al. [118]	Natural Language interface to Relational Database	Semantic and machine learning,	89.79%

No	Author	NLIDB System Name	Method	Accuracy (for reference)
4	W.Wang et al. [213]	NLI Based on Layered Architecture	Sequence-to-Sequence	56.4%
5	T. Mvumbi [153]	NALI System	Top-down, bottom-up Simplified customization Approach	74.5%
6	B.Sujatha et al. [163]	Ontology-Based NLI to Relations DBs	Top-down syntactic parser	84%
7	R.Kokare et al. [133]	Query Builder Based on Dependency Parsing	Dependency Parsing	91.66%
8	L.Safari et al. [123]	Restricted NL Querying of Clinical DBs	A special-purpose clinical data analytics language (CliniDAL)	84%
9	M.Llopis et al. [96]	AskME System	AskME	94.8%
10	U. P. Gunjal et al. [173]	Semantic Grammar System Intelligent System for Relational DBs	Semantic matching technique	75%
11	N. Sukthankar et al. [186]	Pattern-Matching System nQuery System	Pattern-Matching	86%
12	K.Shabaz et al. [140]	Aneesah System	pattern matching technique	85.01%
13	Wenlu Wang et al. [214]	System Based on Other Approaches Transfer-Learnable NLIDB	Sequence-to-Sequence	82%
14	G. et al. [126]	NLI Based on Semantic Representations	using Ontologies semantic modelling,	85.2%
15	Y.-J. Han et al. [149]	NLI Concordant With Knowledge Base	a graph-based knowledge base	86.6%
16	A. Giordani et al. [72]	Question translation with generative parser	grammatical dependencies and metadata matching	85%
17	R. Nichante et al. [180]	System for languages different from English Hindi NLIDB		Not measured
18	H.Bais et al. [144]	GANLIDB System	machine learning and natural language processing	95.1%
19	D.T.Nguyen et al. [80]	Vietnamese NLIDB	GATE framework and R language.	78.5%

No	Author	NLIDB System Name	Method	Accuracy (for reference)
20	Ani Anisyah et al. [224]	Decision support queries for Indonesian language	PC-PATR syntactic parser	Not measured
21	G´amez, Ismael Esquive et al. [366]	Spanish NLIDB	Statistical Expression	76.46%

Chapter 5

Conclusion

This work surveyed the current state of Afghan NLP, NLIDB systems, and proposed a prototype of an Afghan NLIDB system. The survey of Afghan NLP contributes to supplementary and additional Afghan NLP applications. The spoken natural language interface to databases using the Afghan language was built to overcome accessibility challenges as nowadays databases are essential and have the high utilization in government, organizations, hospitals, businesses, research etc. To retrieve data from a database, knowledge of Structured Query Language or a well-designed user interface is required. For most of the end users, both are inadequate. There is a shortage of people who can work on developing interface to access datasets and databases. To solve these gaps, we build a natural language interface for the Afghan language. The Afghan language is one of the most widely used languages, with up to 110 million speakers worldwide. This application is designed for Persian speakers to retrieve data using the Persian language as input. Afterwards the process converts it to the corresponding SQL Query for specific purposes like selecting data, counting records, deleting records, etc. The system makes the use of databases accessible and user-friendly for non-expert domain users who are speaking Persian language around the world. In our work, users input questions in the Afghan language, after input, we used Google Translate for translating Afghan sentences to corresponding English sentences. Then tokenization is implemented on translating sentences. After tokenization, we applied Syntax analysis like stemming and lemmatization. In the semantic analysis phase, we applied part of speech tagging after semantic analyses the final SQL Query is generated, and the final result is shown to the user in the graphical user interface. The environment created through dialogue-based processes like question-answering systems. The mentioned system is very user-friendly and easy to use. The Persian natural language interface to the database, for now, deals with “SELECT”, “FROM”, “WHERE”, “DELETE”. In the future, we want to work on a complex query like “GROUP BY”, “HAVING”, which is used to access a lot of tables for retrieving information to the user. GROUP BY is used to show the result of the similar rows in data into groups, the HAVING clause is used in SQL to retrieve data with Aggregate function because the WHERE clause can not do that.

Future Work: Effective NLIDB system requires efficient transportation. As a research avenue for future work in Afghan NLP, the machine translation of Dari to English is a priority area. Because some words that exist in Dari do not exist in Persian or Persian-English, MT. It would require building the Dari-English Parallel corpus. For complex future and advanced work query, more work needs to be done on both the how to form appropriate speech input and then how to translate that to respective complex SQL query while minimizing the ambiguity. Supplementing the NLIDB system with Asterisk IVR to make the system accessible from land phones and feature phones instead of just online using PC or smartphone remains a goal.

Based on the survey of Afghan NLP, further research areas can be ventured into. examples are spam and fake news detection, information extraction, dependency parsing, etc.

Bibliography

[1] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[2] T. Winograd, "Shrdlu: A system for dialog," 1972.

[3] G. G. Hendrix, E. D. Sacerdoti, D. Sagalowicz, and J. Slocum, "Developing a natural language interface to complex data," *ACM Transactions on Database Systems (TODS)*, vol. 3, no. 2, pp. 105–147, 1978.

[4] C. T. Hemphill, J. J. Godfrey, and G. R. Doddington, "The ATIS spoken language systems pilot corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*, 1990. [Online]. Available: https://aclanthology.org/H90-1021.

[5] G. Hirst, *Semantic interpretation and the resolution of ambiguity.* Cambridge University Press, 1992.

[6] I. Androutsopoulos, G. D. Ritchie, and P. Thanisch, "Natural language interfaces to databases–an introduction," *Natural language engineering*, vol. 1, no. 1, pp. 29–81, 1995.

[7] J. W. Amtrup and H. M. Rad, "Persian-english machine translation: An overview of the shiraz project," *Memoranda in Computer and Cognitive Science MCCS-00-319, NMSU, CRL*, 2000.

[8] X. Meng and S. Wang, "Nchiql: The chinese natural language interface to databases," in *International Conference on Database and Expert Systems Applications*, Springer, 2001, pp. 145–154.

[9] R. A. Pazos Range, A. Gelbukh, J. Barbosa, E. A. Ruiz, A. M. Mejía, and A. Sánchez, "Spanish natural language interface for a relational database querying system," in *International Conference on Text, Speech and Dialogue*, Springer, 2002, pp. 123–130.

[10] J. W. Amtrup, "Morphology in machine translation systems: Efficient integration of finite state transducers and feature structure descriptions," *Machine Translation*, vol. 18, no. 3, pp. 217–238, 2003.

[11] N. Srinivasamurthy and S. Narayanan, "Language-adaptive persian speech recognition," in *Eighth European Conference on Speech Communication and Technology*, 2003.

[12] S. Arsov, "An approach for response generation of restricted bulgarian natural language queries.," in *CompSysTech*, Citeseer, 2004, pp. 1–6.

[13] H. Feili and G. Ghassem-Sani, “An application of lexicalized grammars in english-persian translation,” in *ECAI*, Citeseer, vol. 16, 2004, p. 596.

[14] M. Hassel and N. Mazdak, “Farsisum-a persian text summarizer,” in *Proceedings of the workshop on computational approaches to Arabic script-based languages*, 2004, pp. 82–84.

[15] M. Abdolahi and H. Amindavar, “Garch coefficients as feature for speech recognition in persian isolated digit,” in *Proceedings.(ICASSP’05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, IEEE, vol. 1, 2005, pp. I–957.

[16] T. M. Miangah and A. D. Khalafi, “Word sense disambiguation using target language corpus in a machine translation system,” *Literary and Linguistic Computing*, vol. 20, no. 2, pp. 237–249, 2005.

[17] B. QasemiZadeh and S. Rahimi, “Persian in multext-east framework,” in *International Conference on Natural Language Processing (in Finland)*, Springer, 2006, pp. 541–551.

[18] H. Veisi, H. Sameti, B. Babaali, K. Hosseinzadeh, and M. Manzuri, “Improving the robustness of persian large vocabulary continuous speech recognition system for real applications,” in *2006 2nd International Conference on Information & Communication Technologies*, IEEE, vol. 1, 2006, pp. 1293–1297.

[19] S. Ahadi, “Improved bayesian training for context-dependent modeling in continuous persian speech recognition,” *Journal of Iranian Association of Electrical and Electronics Engineers*, vol. 4, no. 1, pp. 20–26, 2007.

[20] M. Nazari and K. Faez, *Speaker detection and clustering with svm technique in persian conversational speech*, 2007.

[21] F. Raja and S. Tasharofi, “Statistical pos tagging experiments on persian text,” 2007.

[22] H. Faili, “An experiment of word sense disambiguation in a machine translation system,” in *2008 International Conference on Natural Language Processing and Knowledge Engineering*, IEEE, 2008, pp. 1–7.

[23] H. Hasanabadi, A. Rowhanimanesh, and Yazdi, “A simple and robust persian speech recognition system and its application to robotics,” in *2008 International Conference on Advanced Computer Theory and Engineering*, IEEE, 2008, pp. 239–245.

[24] M. Mohseni, H. Motalebi, B. Minaei-Bidgoli, and M. Shokrollahi-far, “A farsi part-of-speech tagger based on markov model,” in *Proceedings of the 2008 ACM symposium on Applied computing*, 2008, pp. 1588–1589.

[25] M. Mohtarami and H. Amiri, “Using heuristic rules to improve persian part of speech tagging accuracy,” 2008.

[26] M. Nazari, A. Sayadiyan, and S. M. Valiollahzadeh, “Speaker-independent vowel recognition in persian speech,” in *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications*, IEEE, 2008, pp. 1–5.

[27] M. Norouzifard, S. Davarpanah, M. Shenassa, *et al.*, “Using natural language processing in order to create sql queries,” in *2008 International Conference on Computer and Communication Engineering*, IEEE, 2008, pp. 600–604.

[28] R. Sarikaya, Y. Deng, M. Afify, B. Kingsbury, and Y. Gao, “Machine translation in continuous space,” Sep. 2008, pp. 2350–2353. DOI: 10.21437/Interspeech.2008-599.

[29] A. Zamanifar, B. Minaei-Bidgoli, and M. Sharifi, “A new hybrid farsi text summarization technique based on term co-occurrence and conceptual property of the text,” in *2008 Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, IEEE, 2008, pp. 635–639.

[30] M. Berenjkoob and Mehri, “A method for stemming and eliminating common words for persian text summarization,” in *2009 International Conference on Natural Language Processing and Knowledge Engineering*, IEEE, 2009, pp. 1–6.

[31] A. K. Ghalibaf, S. Rahati, and A. Estaji, “Shallow semantic parsing of persian sentences,” in *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, 2009, pp. 150–159.

[32] S. Kiani and Akhavan, “Developing a persian chunker using a hybrid approach,” in *2009 International Multiconference on Computer Science and Information Technology*, IEEE, 2009, pp. 227–234.

[33] M. Mohaghegh and A. Sarrafzadeh, “An analysis of the effect of training data variation in english-persian statistical machine translation,” in *2009 International Conference on Innovations in Information Technology (IIT)*, IEEE, 2009, pp. 105–109.

[34] M. Mohamadpour and F. Farokhi, “A new approach for persian speech recognition,” in *2009 IEEE international advance computing conference*, IEEE, 2009, pp. 153–158.

[35] Y. Motazedi and M. Shamsfard, “English to persian machine translation exploiting semantic word sense disambiguation,” in *2009 14th International CSI Computer Conference*, IEEE, 2009, pp. 253–258.

[36] C. Saedi, M. Shamsfard, and Y. Motazedi, “Automatic translation between english and persian texts,” in *Proceedings of the Third Workshop on Computational Approaches to Arabic-Script-based Languages (CAASL3)*, 2009.

[37] M. Shahabi, “An evaluation of output quality of machine translation program,” in *Proceedings of the Student Research Workshop*, 2009, pp. 71–75.

[38] M. Shamsfard, T. Akhavan, and M. E. Jourabchi, “Parsumist: A persian text summarizer,” in *2009 International Conference on Natural Language Processing and Knowledge Engineering*, IEEE, 2009, pp. 1–7.

[39] S. Bakhshaei, S. Khadivi, and N. Riahi, “Farsi-german statistical machine translation through bridge language,” in *2010 5th International Symposium on Telecommunications*, IEEE, 2010, pp. 557–561.

[40] M. Bokaei, H. Sameti, and Eghbal-zadeh, “Niusha, the first persian speech-enabled ivr platform,” in *2010 5th International Symposium on Telecommunications*, IEEE, 2010, pp. 591–595.

[41] M. Mohaghegh, A. Sarrafzadeh, and T. Moir, “Improved language modeling for english-persian statistical machine translation,” in *Proceedings of the 4th Workshop on Syntax and Structure in Statistical Translation*, 2010, pp. 75–82.

[42] M. Mohseni and B. Minaei-Bidgoli, “A persian part-of-speech tagger based on morphological analysis,” in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10)*, 2010.

[43] M. T. Pilevar and H. Faili, “Persiansmt: A first attempt to english-persian statistical machine translation,” in *JADT*, vol. 2010, 2010, 10th.

[44] M. Soltani and H. Faili, “A statistical approach on persian word sense disambiguation,” in *2010 The 7th International Conference on Informatics and Systems (INFOS)*, IEEE, 2010, pp. 1–6.

[45] M. Bahrani and Sameti, “A computational grammar for persian based on gpsg,” *Language Resources and Evaluation*, vol. 45, no. 4, pp. 387–408, 2011.

[46] S. Fakhrahmad, A. Rezapour, and ZolghadriJahromi, “A new word sense disambiguation system based on deduction,” in *Proceedings of the World Congress on Engineering*, vol. 2, 2011, pp. 1276–1281.

[47] M. Haghollahi and M. Shamsfard, “A semi-supervised approach for key-synset extraction to be used in word sense disambiguation,” in *Asia Information Retrieval Symposium*, Springer, 2011, pp. 594–603.

[48] A. A. Hazbavi, “Investigating the effect of translation memory on english into persian translation,” *World Applied Sciences Journal*, vol. 15, no. 5, pp. 683–689, 2011.

[49] H. Kamyar and M. Kahani, “An automatic linguistics approach for persian document summarization,” in *2011 International Conference on Asian Language Processing*, IEEE, 2011, pp. 141–144.

[50] F. Kiyomarsi and F. R. Esfahani, “Optimizing persian text summarization based on fuzzy logic approach,” in *2011 international conference on intelligent building and management*, 2011.

[51] Z. Komeili, A. Rahimi, *et al.*, “An investigation of the translation problems incurred by english-to-persian machine translations:“padideh, pars, and google softwares”,” *Procedia-Social and Behavioral Sciences*, vol. 28, pp. 1079–1082, 2011.

[52] M. Mohaghegh, A. Sarrafzadeh, and T. Moir, "Improving persian-english statistical machine translation: Experiments in domain adaption," 2011.

[53] M. Montazery and H. Faili, "Unsupervised learning for persian wordnet construction," in *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, 2011, pp. 302–308.

[54] M. Nameh, S. Fakhrahmad, and M. Z. Jahromi, "A new approach to word sense disambiguation based on context similarity," in *Proceedings of the World Congress on Engineering*, vol. 1, 2011, pp. 6–8.

[55] M. Okhovvat and B. M. Bidgoli, "A hidden markov model for persian part-of-speech tagging," *Procedia Computer Science*, vol. 3, pp. 977–981, 2011.

[56] A. H. Pilevar, "Using statistical post-editing to improve the output of rule-based machine translation system," *Training*, vol. 330, pp. 330–000, 2011.

[57] A. Poormasoomi and M. Kahani, "Context-based persian multi-document summarization (global view)," in *2011 International Conference on Asian Language Processing*, IEEE, 2011, pp. 145–149.

[58] H. Sameti and H. Veisi, "A large vocabulary continuous speech recognition system for persian language," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2011, no. 1, pp. 1–12, 2011.

[59] B. Sarrafzadeh and Yakovets, "Cross-lingual word sense disambiguation for languages with scarce resources," in *Canadian Conference on Artificial Intelligence*, Springer, 2011, pp. 347–358.

[60] B. Sarrafzadeh and Yakovets, "Towards automatic acquisition of a fully sense tagged corpus for persian," in *International Symposium on Methodologies for Intelligent Systems*, Springer, 2011, pp. 449–455.

[61] M. Seraji, "A statistical part-of-speech tagger for persian," in *NODALIDA 2011, Riga, Latvia, May 11–13, 2011*, 2011, pp. 340–343.

[62] G. T. Tabrizi, S. Setayeshi, and M. M. Kakhki, "Hmm-based recognition and adaptation of persian children's speech," *Department of Computer, Science, and Research Branch*, vol. 4, no. 5, pp. 221–228, 2011.

[63] A. Tavanaei, M. T. Manzuri, and H. Sameti, "Mel-scaled discrete wavelet transform and dynamic features for the persian phoneme recognition," in *2011 international symposium on artificial intelligence and signal processing (AISP)*, IEEE, 2011, pp. 138–140.

[64] M. Tofighy and Kashefi, "Persian text summarization using fractal theory," in *International conference on informatics engineering and information science*, Springer, 2011, pp. 651–662.

[65] A. Zamanifar and O. Kashefi, "Azom: A persian structured text summarizer," in *International Conference on Application of Natural Language to Information Systems*, Springer, 2011, pp. 234–237.

[66] A. A. Ahangar, N. Jahangiri, and F. Mohammadpour, "A lexical-functional model for machine translation of english zero-place predicators into persian," *International Journal of English Linguistics*, vol. 2, no. 3, p. 2, 2012.

[67] S. Bakhshaei and S. Khadivi, "A pool-based active learning method for improving farsi-english machine translation system," in *6th International Symposium on Telecommunications (IST)*, IEEE, 2012, pp. 822–826.

[68] M. Bazghandi and Tabrizi, "Extractive summarization of farsi documents based on pso clustering," *jiA*, vol. 1, p. 1, 2012.

[69] M. Berenjkoub and M. Palhang, "Persian text summarization using a supervised machine learning approach," in *Proceedings of the Robocup IranOpen Symposium and 2nd Irans Joint Conference of Robotics and AI Tehran Iran*, 2012.

[70] C. I. Davis, "Tajik-farsi persian transliteration using statistical machine translation," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, 2012, pp. 3988–3995.

[71] M. Ghayoomi, "Word clustering for persian statistical parsing," in *International Conference on NLP*, Springer, 2012, pp. 126–137.

[72] A. Giordani and A. Moschitti, "Translating questions to sql queries with generative parsers discriminatively reranked," in *Proceedings of COLING 2012: Posters*, 2012, pp. 401–410.

[73] M. M. Goodarzi and Almasganj, "Feature bandwidth extension for persian conversational telephone speech recognition," in *20th Iranian Conference on Electrical Engineering (ICEE2012)*, IEEE, 2012, pp. 1220–1223.

[74] M. Hamidi and M. Mansoorizade, "Emotion recognition from persian speech with neural network," *International Journal of Artificial Intelligence & Applications*, vol. 3, no. 5, p. 107, 2012.

[75] F. Jabbari, S. Bakshaei, S. M. M. Ziabary, and S. Khadivi, "Developing an open-domain english-farsi translation system using afec: Amirkabir bilingual farsi-english corpus," in *Fourth Workshop on Computational Approaches to Arabic-Script-based Languages*, 2012, pp. 17–23.

[76] F. Jani and A. Pilevar, "Word sense disambiguation of persian homographs.," in *ICSOFT*, 2012, pp. 328–331.

[77] A. Mansouri and H. Faili, "State-of-the-art english to persian statistical machine translation system," in *The 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012)*, IEEE, 2012, pp. 174–179.

[78] M. Mohaghegh, "Advancements in english-persian hierarchical statistical machine translation," 2012.

[79] A. Mollaei, S. Rahati-Quchani, and A. Estaji, "Question classification in persian language based on conditional random fields," in *2012 2nd International eConference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2012, pp. 295–300.

[80] D. T. Nguyen, T. D. Hoang, and S. B. Pham, “A vietnamese natural language interface to database,” in *2012 IEEE Sixth International Conference on Semantic Computing*, IEEE, 2012, pp. 130–133.

[81] D. T. Nguyen, T. D. Hoang, and S. B. Pham, “A vietnamese natural language interface to database,” in *2012 IEEE Sixth International Conference on Semantic Computing*, IEEE, 2012, pp. 130–133.

[82] S. Pirhosseinloo and F. A. Ganj, “Discriminative speaker adaptation in persian continuous speech recognition systems,” *Procedia-Social and Behavioral Sciences*, vol. 32, pp. 296–301, 2012.

[83] N. Riahi and F. Sedghi, “A semi-supervised method for persian homograph disambiguation,” in *20th Iranian Conference on Electrical Engineering (ICEE2012)*, IEEE, 2012, pp. 748–751.

[84] P. Saeedi and H. Faili, “Feature engineering using shallow parsing in argument classification of persian verbs,” in *The 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012)*, IEEE, 2012, pp. 333–338.

[85] M. Seraji and Megyesi, “Bootstrapping a persian dependency treebank,” *Linguistic Issues in Language Technology*, vol. 7, no. 18, 2012.

[86] H. Shakeri and Gholamrezazadeh, “A new graph-based algorithm for persian text summarization,” in *Computer science and convergence*, Springer, 2012, pp. 21–30.

[87] B. Sujatha, D. S. V. Raju, and H. Shaziya, “A survey of natural language interface to database management system,” *International Journal of Science and Advance Technology*, vol. 2, no. 6, pp. 56–61, 2012.

[88] H. Azarbonyad, A. Shakery, and H. Faili, “Exploiting multiple translation resources for english-persian cross language information retrieval,” in *International Conference of the Cross-Language Evaluation Forum for European Languages*, Springer, 2013, pp. 93–99.

[89] A. Bagheri, M. Saraee, and F. de Jong, “Sentiment classification in persian: Introducing a mutual information-based method for feature selection,” in *2013 21st Iranian conference on electrical engineering (ICEE)*, IEEE, 2013, pp. 1–6.

[90] N. Bertoldi and Farajian, “Fbk’s machine translation systems for the iwslt 2013 evaluation campaign,” in *Proceedings of the 10th International Workshop on Spoken Language Translation: Evaluation Campaign*, 2013.

[91] N. Bertoldi, M. A. Farajian, P. Mathur, N. Ruiz, and M. Federico, “Fbk’s machine translation systems for the iwslt 2013 evaluation campaign,” in *Proceedings of the 10th International Workshop on Spoken Language Translation: Evaluation Campaign*, 2013.

[92] N. Ehsan and H. Faili, “Grammatical and context-sensitive error correction using a statistical machine translation framework,” *Software: Practice and Experience*, vol. 43, no. 2, pp. 187–206, 2013.

[93] A. El Kholy and Habash, "Language independent connectivity strength features for phrase pivot statistical machine translation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2013, pp. 412–418.

[94] A. El Kholy and Habash, "Selective combination of pivot and direct statistical machine translation models," in *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, 2013, pp. 1174–1180.

[95] A. Kumar and K. Singh, "Hindi language interface to database using semantic matching," *Oriental Journal of Computer Science & Technology*, vol. 6, no. 2, pp. 133–140, 2013.

[96] M. Llopis and A. Ferrández, "How to make a natural language interface to query databases accessible to everyone: An example," *Computer Standards & Interfaces*, vol. 35, no. 5, pp. 470–481, 2013.

[97] A. Mahmoudi, H. Faili, and M. Arabsorkhi, "Modeling persian verb morphology to improve english-persian machine translation," in *Mexican International Conference on Artificial Intelligence*, Springer, 2013, pp. 406–418.

[98] M. R. Mirsarraf and N. Dehghani, "A dependency-inspired semantic evaluation of machine translation systems," in *International conference of the cross-language evaluation forum for European languages*, Springer, 2013, pp. 71–74.

[99] B. Mirzababaei, H. Faili, and N. Ehsan, "Discourse-aware statistical machine translation as a context-sensitive spell checker," in *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, 2013, pp. 475–482.

[100] B. B. Moghaddas and Kahani, "Pasokh: A standard corpus for the evaluation of persian text summarizers," in *ICCKE 2013*, IEEE, 2013, pp. 471–475.

[101] M. S. Rasooli, A. El Kholy, and N. Habash, "Orthographic and morphological processing for persian-to-english statistical machine translation," in *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, 2013, pp. 1047–1051.

[102] M. S. Rasooli and Kouhestani, "Development of a persian syntactic dependency treebank," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 306–314.

[103] M. Saraee and A. Bagheri, "Feature selection methods in persian sentiment analysis," in *International conference on application of natural language to information systems*, Springer, 2013, pp. 303–308.

[104] M. Seraji, "Preper: A pre-processor for persian," 2013.

[105] S. M. Tofighy, R. G. Raj, and H. H. S. Javad, "Ahp techniques for persian text summarization," *Malaysian Journal of Computer Science*, vol. 26, no. 1, pp. 1–8, 2013.

[106] E. Ansari and Sadreddini, “Extracting persian-english parallel sentences from document level aligned comparable corpus using bidirectional translation,” *Advances in Computer Science: An International Journal*, vol. 3, no. 5, pp. 59–65, 2014.

[107] A. Bagheri and M. Saraee, “Persian sentiment analyzer: A framework based on a novel feature selection method,” *arXiv preprint arXiv:1412.8079*, 2014.

[108] M. E. Basiri, A. R. Naghsh-Nilchi, and N. Ghassem-Aghaee, “A framework for sentiment analysis in persian,” *Open transactions on information processing*, vol. 1, no. 3, pp. 1–14, 2014.

[109] M. BAZRFKAN and M. RADMANESH, “Using machine learning methods to summarize persian texts,” *Indian J. Sci. Res*, vol. 7, no. 1, pp. 1325–1333, 2014.

[110] M. M. Chauhan and M. N. Bhati, “A noval hindi language interface for databases,” 2014.

[111] W. Feely and Manshadi, “The cmu metal farsi nlp approach,” in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, 2014, pp. 4052–4055.

[112] A. Harimi and Z. Esmaileyan, “A database for automatic persian speech emotion recognition: Collection, processing and evaluation,” *International Journal of Engineering*, vol. 27, no. 1, pp. 79–90, 2014.

[113] S. M. Hoseini and A. A. Azirani, “Statistical language model adaptation for persian speech recognition,” 2014.

[114] S. A. Hosseinzadeh, “Application of machine learning techniques in persian text summarization systems,” 2014.

[115] S. Jindal, M. Dua, and Z. S. Virk, “Khik: Karaka-based hindi language interface to knowledge base,” in *Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies*, 2014, pp. 1–5.

[116] A. A. Kardan and M. B. Imani, “Improving persian pos tagging using the maximum entropy model,” in *2014 Iranian Conference on Intelligent Systems (ICIS)*, IEEE, 2014, pp. 1–5.

[117] M. K. Khormuji and M. Bazrafkan, “Persian named entity recognition based with local filters,” *International Journal of Computer Applications*, vol. 100, no. 4, 2014.

[118] F. Li and H. V. Jagadish, “Nalir: An interactive natural language interface for querying relational databases,” in *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, 2014, pp. 709–712.

[119] E. Mahdipour and M. Bagheri, “Automatic persian text summarizer using simulated annealing and genetic algorithm,” *International Journal of Intelligent Information Systems, Special Issue: Research and Practices in Information Systems and Technologies in Developing Countries*, vol. 3, no. 6-1, pp. 84–90, 2014.

[120] A. POORMASOOMI and KAHANI, “Ijaz: An operational system for single-document summarization of persian news texts,” 2014.

[121] E. Rabbani and A. Shakery, “Analyzing content-based heuristics for persian web spam detection,” 2014.

[122] A. Rezapour and Fakhrahmad, “An accurate word sense disambiguation system based on weighted lexical features,” *Literary and Linguistic Computing*, vol. 29, no. 1, pp. 74–88, 2014.

[123] L. Safari and J. D. Patrick, “Restricted natural language based querying of clinical databases,” *Journal of biomedical Informatics*, vol. 52, pp. 338–353, 2014.

[124] E. Sherkat and M. Farhoodi, “A hybrid approach for question classification in persian automatic question answering systems,” in *2014 4th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2014, pp. 279–284.

[125] F. Ahmadi and H. Moradi, “A hybrid method for persian named entity recognition,” in *2015 7th conference on information and knowledge technology (IKT)*, IEEE, 2015, pp. 1–7.

[126] G. et al., “Semantic representations for knowledge modelling of a natural language interface to databases using ontologies,” *International Journal of Combinatorial Optimization Problems and Informatics*, vol. 6, no. 2, pp. 28–42, 2015.

[127] M. Alimohammadi and M. Zahedi, “Communication between deaf and hearing children using statistical machine translation,” in *2015 2nd International Conference on Pattern Recognition and Image Analysis (IPRIA)*, IEEE, 2015, pp. 1–6.

[128] F. Amiri, S. Scerri, and M. Khodashahi, “Lexicon-based sentiment analysis for persian text,” in *Proceedings of the International Conference Recent Advances in Natural Language Processing*, 2015, pp. 9–16.

[129] D. Darabian, H. Marvi, and M. Sharif Noughabi, “Improving the performance of mfcc for persian robust speech recognition,” *Journal of AI and Data Mining*, vol. 3, no. 2, pp. 149–156, 2015.

[130] K. N. ElSayed, “An arabic natural language interface system for a database of the holy quran,” *International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 7, pp. 9–14, 2015.

[131] S. Farzi, H. Faili, and S. Khadivi, “A syntactically informed reordering model for statistical machine translation,” *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 27, no. 4, pp. 449–469, 2015.

[132] N. Keshtiari, M. Kuhlmann, M. Eslami, and G. Klann-Delius, “Recognizing emotional speech in persian: A validated database of persian emotional speech (persian esd),” *Behavior research methods*, vol. 47, no. 1, pp. 275–294, 2015.

[133] R. Kokare and K. Wanjale, "A natural language query builder interface for structured databases using dependency parsing," *International Journal of Mathematical Sciences and Computing*, vol. 1, no. 4, pp. 11–20, 2015.

[134] M. Mahmoodvand and M. Hourali, "Persian word sense disambiguation corpus extraction based on web crawler method," *Advances in Computer Science: an International Journal*, vol. 4, no. 5, pp. 101–106, 2015.

[135] M. Narouei, M. Ahmadi, and A. Sami, "Sepas: Word sense disambiguation by sequential patterns in sentences," *Natural Language Engineering*, vol. 21, no. 2, pp. 251–269, 2015.

[136] A. Nourian, M. S. Rasooli, M. Imany, and H. Faili, "On the importance of ezafe construction in persian parsing," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2015, pp. 877–882.

[137] A. Palakurthi, S. Ruthu, A. Akula, and R. Mamidi, "Classification of attributes in a natural language query into different sql clauses," in *Proceedings of the International Conference Recent Advances in Natural Language Processing*, 2015, pp. 497–506.

[138] M. Savargiv and A. Bastanfard, "Persian speech emotion recognition," in *2015 7th Conference on Information and Knowledge Technology (IKT)*, IEEE, 2015, pp. 1–5.

[139] M. Seraji, B. Bohnet, and J. Nivre, "Parsper: A dependency parser for persian," in *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, 2015, pp. 300–309.

[140] K. Shabaz, J. D. O'Shea, K. A. Crockett, and A. Latham, "Aneesah : A conversational natural language interface to databases," 2015.

[141] A. Zafarian, A. Rokni, and Khadivi, "Semi-supervised learning for named entity recognition using weakly labeled training data," in *2015 the international symposium on artificial intelligence and signal processing (AISP)*, IEEE, 2015, pp. 129–135.

[142] H. Afshin and M. A. Alaeddini, "A contrastive analysis of machine translation (google translate) and human translation: Efficacy in translating verb tense from english to persian," *Mediterranean Journal of Social Sciences*, vol. 7, no. 4 S2, p. 40, 2016.

[143] B. Ahmadnia and J. Serrano, "Direct translation vs. pivot language translation for persian-spanish low-resourced statistical machine translation system," in *Proceedings of the 18th International Conference on Artificial Intelligence and Computer Science*, 2016.

[144] H. Bais, M. Machkour, and L. Koutti, "An independent-domain natural language interface for relational database: Case arabic language," in *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, IEEE, 2016, pp. 1–7.

[145] K. Dashtipour, A. Hussain, and Zhou, “Persent: A freely available persian sentiment lexicon,” in *International conference on brain inspired cognitive systems*, Springer, 2016, pp. 310–320.

[146] H. Faili and Ehsan, “Vafa spell-checker for detecting spelling, grammatical, and real-word errors of persian language,” *Digital Scholarship in the Humanities*, vol. 31, no. 1, pp. 95–117, 2016.

[147] H. Ghasemi and M. Hashemian, “A comparative study of google translate translations: An error analysis of English-to-Persian and Persian-to-English translations,” *Engl. Lang. Teach.*, vol. 9, no. 3, p. 13, 2016.

[148] H. Ghasemi and M. Hashemian, “A comparative study of” google translate” translations: An error analysis of english-to-persian and persian-to-english translations.,” *English Language Teaching*, vol. 9, no. 3, pp. 13–17, 2016.

[149] Y.-J. Han, S.-B. Park, and S.-Y. Park, “A natural language interface concordant with a knowledge base,” *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–15, Jan. 2016. DOI: 10.1155/2016/9174683.

[150] Z. Hosseini Pozveh and Monadjemi, “Persian texts part of speech tagging using artificial neural networks,” *Journal of Computing and Security*, vol. 3, no. 4, pp. 233–241, 2016.

[151] W. Li, “Research on the automatic extraction of persian simple sentences,” in *2016 International Conference on Sensor Network and Computer Engineering*, Atlantis Press, 2016, pp. 77–82.

[152] Z. Mirzanezhad and M.-R. Feizi-Derakhshi, “Using morphological analyzer to statistical pos tagging on persian text,” *Int J Comput Sci Inf Secur (IJCSIS)*, vol. 14, no. 8, 2016.

[153] T. Mvumbi, “Natural language interface to relational database: A simplified customization approach,” M.S. thesis, University of Cape Town, 2016.

[154] A. Pakzad and B. Minaei Bidgoli, “An improved joint model: Pos tagging and dependency parsing,” *Journal of AI and Data Mining*, vol. 4, no. 1, pp. 1–8, 2016.

[155] S. Parvandeh, S. Lahiri, and F. Boroumand, “Persum: Novel systems for document summarization in persian,” *arXiv preprint arXiv:1606.03143*, 2016.

[156] J. Pérez *et al.*, “Comparative study on the customization of natural language interfaces to databases,” *SpringerPlus*, vol. 5, no. 1, pp. 1–30, 2016.

[157] H. Poostchi and Borzeshi, “Personer: Persian named-entity recognition,” in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 3381–3389.

[158] N. Rekabsaz and Sabetghadam, “Standard test collection for english-persian cross-lingual word sense disambiguation,” in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, 2016, pp. 4176–4179.

[159] N. Riahi and F. Sedghi, “Improving the collocation extraction method using an untagged corpus for persian word sense disambiguation,” *Journal of Computer and Communications*, vol. 4, no. 4, pp. 109–124, 2016.

[160] D. Saha, A. Floratou, K. Sankaranarayanan, U. F. Minhas, A. Mittal, and F. Özcan, "Athena: An ontology-driven system for natural language querying over relational data stores," *Proceedings of the VLDB Endowment*, vol. 9, pp. 1209–1220, Aug. 2016. DOI: 10.14778/2994509.2994536.

[161] S. Salami and M. Shamsfard, "Phrase-boundary model for statistical machine translation," *Computer Speech & Language*, vol. 38, pp. 13–27, 2016.

[162] S. Salami, M. Shamsfard, and S. Khadivi, "Phrase-boundary model for statistical machine translation," *Computer Speech & Language*, vol. 38, pp. 13–27, 2016.

[163] B. Sujatha and S. V. Raju, "Ontology based natural language interface for relational databases," *Procedia Computer Science*, vol. 92, pp. 487–492, 2016.

[164] N. Taghizadeh and H. Faili, "Automatic wordnet development for low-resource languages using cross-lingual wsd," *Journal of Artificial Intelligence Research*, vol. 56, pp. 61–87, 2016.

[165] M. Zahedi and A. Arjomandzadeh, "A new model for persian multi-part words edition based on statistical machine translation," *Journal of AI and Data Mining*, vol. 4, pp. 27–34, 2016.

[166] B. Ahmadnia and J. Serrano, "Employing pivot language technique through statistical and neural machine translation frameworks: The case of under-resourced persian-spanish language pair," *International Journal on Natural Language Computing*, vol. 6, no. 5, pp. 37–47, 2017.

[167] M. E. Basiri and A. Kabiri, "Sentence-level sentiment analysis in persian," in *2017 3rd International Conference on Pattern Recognition and Image Analysis (IPRIA)*, IEEE, 2017, pp. 84–89.

[168] M. E. Basiri and A. Kabiri, "Translation is not enough: Comparing lexicon-based methods for sentiment analysis in persian," in *2017 International Symposium on Computer Science and Software Engineering Conference (CSSE)*, IEEE, 2017, pp. 36–41.

[169] M. Bastan, S. Khadivi, and M. M. Homayounpour, "Neural machine translation on scarce-resource condition: A case-study on persian-english," in *2017 Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2017, pp. 1485–1490.

[170] K. Dashtipour and M. Gogate, "A comparative study of persian sentiment analysis based on different feature combinations," in *International conference in communications, signal processing, and systems*, Springer, 2017, pp. 2288–2294.

[171] K. Dashtipour, M. Gogate, and Adeel, "Persian named entity recognition," in *2017 IEEE 16th international conference on cognitive informatics & cognitive computing (ICCI* CC)*, IEEE, 2017, pp. 79–83.

[172] F. Ebrahimi Rashed and N. Abdolvand, "A supervised method for constructing sentiment lexicon in persian language," *Journal of Computer & Robotics*, vol. 10, no. 1, 2017.

[173] U. P. Gunjal, V. Rathod, and N. N. Pise, “An intelligent system for relational databases.,” *International Journal of Scientific Research (Ahmedabad, India)*, vol. 6, no. 3, pp. 1546–1550, 2017.

[174] S. Hosseinnejad, Y. Shekofteh, and T. Emami Azadi, “A’laam corpus: A standard corpus of named entity for persian language,” *Signal and Data Processing*, vol. 14, no. 3, pp. 127–142, 2017.

[175] M. Mahmoodvand and M. Hourali, “Semi-supervised approach for persian word sense disambiguation,” in *2017 7th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2017, pp. 104–110.

[176] M. M. Mahsuli and R. Safabakhsh, “English to persian transliteration using attention-based approach in deep learning,” in *2017 Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2017, pp. 174–178.

[177] M. Maleki and H. H. Tabrizi, “A corpus-based study of human-translated vs. machine-translated texts: The case of ellipsis in english-persian translation,” *Journal of Applied Linguistics and Language Research*, vol. 4, no. 1, pp. 290–303, 2017.

[178] P. Malhotra, Y. Kapadia, K. Saboo, and A. Sarda, “Question-answering system using natural language processing with nlidb approach,” *International Journal of Current Research*, vol. 9, no. 09, pp. 57 575–57 577, 2017.

[179] N. Mostofian, “A study on manual and automatic evaluation procedures and production of automatic post-editing rules for persian machine translation,” Ph.D. dissertation, Jun. 2017.

[180] R. Nichante, S. Giripunje, A. Nikam, S. Arsod, and N. Sonwane, “Hindi language as a graphical user interface to relational database for transport system,” *International Research Journal of Engineering and Technology*, vol. 4, no. 3, pp. 349–353, 2017.

[181] R. Nichante, S. Giripunje, A. Nikam, S. Arsod, and N. Sonwane, “Hindi language as a graphical user interface to relational database for transport system,” *International Research Journal of Engineering and Technology*, vol. 4, no. 3, pp. 349–353, 2017.

[182] N. Rekabsaz, M. Lupu, A. Hanbury, and A. Duque, “Addressing cross-lingual word sense disambiguation on low-density languages: Application to persian,” *arXiv preprint arXiv:1711.06196*, 2017.

[183] M. J. Rezai and T. Mosavi Miangah, “Farsitag: A part-of-speech tagging system for persian,” *Digital Scholarship in the Humanities*, vol. 32, no. 3, pp. 632–642, 2017.

[184] B. Roshanfekr, S. Khadivi, and M. Rahmati, “Sentiment analysis using deep learning on persian texts,” in *2017 Iranian conference on electrical engineering (ICEE)*, IEEE, 2017, pp. 1503–1508.

[185] H. Shahverdian and H. Saneifar, “Text summarization of multi-aspect comments in social networks in persian language,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 12, 2017.

[186] N. Sukthankar, S. Maharnawar, P. Deshmukh, Y. Haribhakta, and V. Kamble, “NQuery - a natural language statement to SQL query generator,” in *Proceedings of ACL 2017, Student Research Workshop*, Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 17–23. [Online]. Available: https://aclanthology.org/P17-3004.

[187] S. Zamani, M. Asadpour, and D. Moazzami, “Rumor detection for persian tweets,” in *2017 Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2017, pp. 1532–1536.

[188] M. Zand Rahimi, M. Madayenzadeh, and M. Alizadeh, “A comparative study of english-persian translation of neural google translation,” *Iranian Journal of Applied Language Studies*, vol. 9, no. Proceedings of the First International Conference on Language Focus, pp. 279–286, 2017.

[189] P. Zaremoodi and G. Haffari, “Incorporating syntactic uncertainty in neural machine translation with forest-to-sequence model,” *arXiv preprint arXiv:1711.07019*, 2017.

[190] E. Alayiaboozar, “Introducing a machine-based approach for word sense disambiguation: Using lesk algorithm and part of speech tagging,” *Iranian Journal of Information processing and Management*, vol. 33, no. 3, pp. 1165–1182, 2018.

[191] H. Bais, M. Machkour, and L. Koutti, “An arabic natural language interface for querying relational databases based on natural language processing and graph theory methods,” *International Journal of Reasoning-Based Intelligent Systems*, vol. 10, no. 2, pp. 155–165, 2018.

[192] M. E. Basiri and A. Kabiri, “Words are important: Improving sentiment analysis in the persian language by lexicon refining,” *ACM transactions on Asian and low-resource language information processing (TALLIP)*, vol. 17, no. 4, pp. 1–18, 2018.

[193] M. H. Bokaei and M. Mahmoudi, “Improved deep persian named entity recognition,” in *2018 9th International Symposium on Telecommunications (IST)*, IEEE, 2018, pp. 381–386.

[194] Y. Boreshban, “Providing a religious corpus of question answering system in persian,” *Signal and Data Processing*, vol. 15, no. 1, pp. 87–102, 2018.

[195] D. B. BOTSHEKANAN and M. E. BASIRI, “Improving persian sentiment analysis using opposing polarity phrases,” 2018.

[196] K. Dashtipour and M. Gogate, “Exploiting deep learning for persian sentiment analysis,” in *International conference on brain inspired cognitive systems*, Springer, 2018, pp. 597–604.

[197] M. H. Dehghan and Molla-Abbasi, “Toward a multi-representation persian treebank,” in *2018 9th International Symposium on Telecommunications (IST)*, IEEE, 2018, pp. 581–586.

[198] H. Ghassani and T. E. Widagdo, “Access to relational databases using interrogative sentences in indonesian language,” in *2018 5th International Conference on Data and Software Engineering (ICoDSE)*, IEEE, 2018, pp. 1–6.

[199] L. Hafezi and M. Rezaeian, “Neural architecture for persian named entity recognition,” in *2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, IEEE, 2018, pp. 61–64.

[200] P. Hosseini and A. A. Ramaki, “Sentipers: A sentiment analysis corpus for persian,” *arXiv preprint arXiv:1801.07737*, 2018.

[201] T. Hosseinikhah, A. Ahmadi, and A. Mohebi, “A new persian text summarization approach based on natural language processing and graph similarity,” *Iranian Journal of Information Processing and Management*, 2018.

[202] Y. Li and D. Rafiei, *Natural Language Data Management and Interfaces.* San Rafael, CA: Morgan & Claypool, 2018.

[203] S. Mohtaj and B. Roshanfekr, “Parsivar: A language processing toolkit for persian,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[204] D. Q. Nguyen and K. Verspoor, “An improved neural network model for joint pos tagging and dependency parsing,” *arXiv preprint arXiv:1807.03955*, 2018.

[205] H. Poostchi, E. Z. Borzeshi, and M. Piccardi, “Bilstm-crf for persian named-entity recognition armanpersonercorpus: The first entity-annotated persian dataset.,” in *LREC*, 2018.

[206] M. Rahat and A. Talebpour, “A recursive algorithm for open information extraction from persian texts,” *International Journal of Computer Applications in Technology*, vol. 57, no. 3, pp. 193–206, 2018.

[207] M. Rahat and A. Talebpour, “Open information extraction as an intermediate semantic structure for persian text summarization,” *International Journal on Digital Libraries*, vol. 19, no. 4, pp. 339–352, 2018.

[208] M. Rahat and A. Talebpour, “Parsa: An open information extraction system for persian,” *Digital Scholarship in the Humanities*, vol. 33, no. 4, pp. 874–893, 2018.

[209] N. Rasouli, L. Abedi, and S. Ghaei, “Designing an agent for information extraction from persian e-shops,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 1, pp. 455–462, 2018.

[210] M. Razzaghnoori and Sajedi, “Question classification in persian using word vectors and frequencies,” *Cognitive Systems Research*, vol. 47, pp. 16–27, 2018.

[211] M. Sadat Shahshahani, M. Mohseni, A. Shakery, and H. Faili, “Peyma: A tagged corpus for persian named entities,” *arXiv e-prints*, arXiv–1801, 2018.

[212] F. Shafiee and M. Shamsfard, “Similarity versus relatedness: A novel approach in extractive persian document summarisation,” *Journal of Information Science*, vol. 44, no. 3, pp. 314–330, 2018.

[213] W. Wang, Y. Tian, H. Xiong, H. Wang, and W.-S. Ku, *A transfer-learnable natural language interface for databases*, 2018. arXiv: 1809.02649 `[cs.AI]`.

[214] W. Wang, Y. Tian, H. Xiong, H. Wang, and W.-S. Ku, “A transfer-learnable natural language interface for databases,” *arXiv preprint arXiv:1809.02649*, 2018.

[215] M. van der Wees, A. Bisazza, and C. Monz, “Evaluation of machine translation performance across multiple genres and languages,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. [Online]. Available: https://aclanthology.org/L18-1604.

[216] M. Zali and M. Firoozbakht, “Named entities recognition and classification system for persian texts based on neural network,” *Iranian Journal of Information processing and Management*, vol. 34, no. 1, pp. 473–486, 2018.

[217] H. Zeinali and H. Sameti, “Deepmine speech processing database: Text-dependent and independent speaker verification and speech recognition in persian and english.,” in *Odyssey*, 2018, pp. 386–392.

[218] B. Ahmadnia and B. J. Dorr, “Augmenting neural machine translation through round-trip training approach,” *Open Computer Science*, vol. 9, no. 1, pp. 268–278, 2019.

[219] B. Ahmadnia, G. Haffari, and J. Serrano, “Round-trip training approach for bilingually low-resource statistical machine translation systems,” *International Journal of Artificial Intelligence*, vol. 17, no. 1, pp. 167–185, 2019.

[220] R. Akhoundzade and K. H. Devin, “Persian sentiment lexicon expansion using unsupervised learning methods,” in *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2019, pp. 461–465.

[221] E. Alayiaboozar and Moloodi, “Word sense disambiguation focusing on pos tag disambiguation in persian,” *International Journal of Information Science and Management (IJISM)*, vol. 17, no. 2, 2019.

[222] A. Anisyah, T. E. Widagdo, and F. N. Azizah, “Natural language interface to database (nlidb) for decision support queries,” in *2019 International Conference on Data and Software Engineering (ICoDSE)*, IEEE, 2019, pp. 1–6.

[223] A. Anisyah, T. E. Widagdo, and F. N. Azizah, “Natural language interface to database (nlidb) for decision support queries,” in *2019 International Conference on Data and Software Engineering (ICoDSE)*, IEEE, 2019, pp. 1–6.

[224] A. Anisyah, T. E. Widagdo, and F. Nur Azizah, “Natural language interface to database (NLIDB) for decision support queries,” in *2019 International Conference on Data and Software Engineering (ICoDSE)*, Pontianak, Indonesia: IEEE, Nov. 2019.

[225] H. Bais and M. Machkour, “Arabic language interface for xml databases,” in *2019 4th World Conference on Complex Systems (WCCS)*, IEEE, 2019, pp. 1–5.

[226] H. Bais and M. Machkour, “Arabic language interface for xml databases,” in *2019 4th World Conference on Complex Systems (WCCS)*, IEEE, 2019, pp. 1–5.

[227] M. E. Basiri, H. KHOSRAVI, *et al.*, “Feature ranking for persian spam review detection,” 2019.

[228] M. E. Basiri, N. Safarian, and H. K. Farsani, “A supervised framework for review spam detection in the persian language,” in *2019 5th International Conference on Web Research (ICWR)*, IEEE, 2019, pp. 203–207.

[229] R. Dehkharghani, “Sentifars: A persian polarity lexicon for sentiment analysis,” *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 19, no. 2, pp. 1–12, 2019.

[230] A. Derakhshan and H. Beigy, “Sentiment analysis on stock social media for stock price movement prediction,” *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 569–578, 2019.

[231] S. Farzi and S. Kianian, “Katibeh: A persian news summarizer using the novel semi-supervised approach,” *Digital Scholarship in the Humanities*, vol. 34, no. 2, pp. 277–289, 2019.

[232] S. Karimi and F. S. Shahrabadi, “Sentiment analysis using bert (pre-training language representations) and deep learning on persian texts,” 2019.

[233] F. H. Kermani and S. Ghanbari, “Extractive persian summarizer for news websites,” in *2019 5th International Conference on Web Research (ICWR)*, IEEE, 2019, pp. 85–89.

[234] A. Kulkarni and A. Shivananda, “Exploring and processing text data,” in *Natural Language Processing Recipes*, Berkeley, CA: Apress, 2019, pp. 37–65.

[235] S. Lazemi and H. Ebrahimpour-Komleh, “Feature engineering in persian dependency parser,” *Journal of AI and Data Mining*, vol. 7, no. 3, pp. 467–474, 2019.

[236] S. Lazemi, H. Ebrahimpour-Komleh, and N. Noroozi, “Improving persian dependency-based semantic role labeling using semantic and structural relations,” in *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, IEEE, 2019, pp. 163–167.

[237] J. A. V. Maldonado and J. A. G. Cuadra, “Natural language interface to database using the dialogflow voice recognition and text conversion api,” in *2019 8th International Conference On Software Process Improvement (CIMPS)*, IEEE, 2019, pp. 1–10.

[238] J. A. V. Maldonado and J. A. G. Cuadra, “Natural language interface to database using the dialogflow voice recognition and text conversion api,” in *2019 8th International Conference On Software Process Improvement (CIMPS)*, IEEE, 2019, pp. 1–10.

[239] K. P. Mandal, P. Mukherjee, B. Chakraborty, and A. Chattopadhyay, “A novel bengali language query processing system (blqps) in medical domain,” *Intelligent Decision Technologies*, vol. 13, no. 2, pp. 177–192, 2019.

[240] K. P. Mandal, P. Mukherjee, B. Chakraborty, and A. Chattopadhyay, “A novel bengali language query processing system (blqps) in medical domain,” *Intelligent Decision Technologies*, vol. 13, no. 2, pp. 177–192, 2019.

[241] O. Mohamad Nezami, P. Jamshid Lou, and M. Karami, “Shemo: A large-scale validated database for persian speech emotion detection,” *Language Resources and Evaluation*, vol. 53, no. 1, pp. 1–16, 2019.

[242] M. Mohseni and A. Tebbifakhr, “Morphobert: A persian ner system with bert and morphological analysis,” in *Proceedings of The First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019-Short Papers*, 2019, pp. 23–30.

[243] B. Moradi, E. Ansari, and Z. Žabokrtskỳ, “Unsupervised word sense disambiguation using word embeddings,” in *2019 25th Conference of Open Innovations Association (FRUCT)*, IEEE, 2019, pp. 228–233.

[244] S. M. H. Mousavi, V. S. Prasath, and S. M. H. Mousavi, “Persian classical music instrument recognition (pcmir) using a novel persian music database,” in *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2019, pp. 122–130.

[245] M. M. Nematollahi and O. R. Marouzi, “A new representation of open information extraction in persian language,” *International Journal of Nonlinear Analysis and Applications*, vol. 10, no. 2, pp. 189–196, 2019.

[246] Z. B. Nezhad and M. A. Deihimi, “A combined deep learning model for persian sentiment analysis,” *IIUM Engineering Journal*, vol. 20, no. 1, pp. 129–139, 2019.

[247] H. Rezaei, S. A. Moeinzadeh, A. Shahgholian, and M. Saraee, “Features in extractive supervised single-document summarization: Case of persian news,” *arXiv preprint arXiv:1909.02776*, 2019.

[248] B. Sabeti and Hosseini, “Lexipers: An ontology-based sentiment lexicon for persian,” *arXiv preprint arXiv:1911.05263*, 2019.

[249] A. R. Shahbazkia, “Machine translation by homograph detector with the help of grammatical base of persian words,” 2019.

[250] T. Shangipour Ataei and K. Darvishi, “Pars-absa: An aspect-based sentiment analysis dataset for persian,” *arXiv e-prints*, arXiv–1908, 2019.

[251] M. Taleghani, E. Pazouki, and V. Ghahraman, “The correlation of machine translation evaluation metrics with human judgement on persian language,” *Journal of Language and Translation*, vol. 9, no. 3, pp. 43–55, 2019.

[252] B. Xu, R. Cai, Z. Zhang, *et al.*, “Nadaq: Natural language database querying based on deep learning,” *IEEE Access*, vol. 7, pp. 35 012–35 017, 2019. DOI: 10.1109/ACCESS.2019.2904720.

[253] T. Yu, R. Zhang, K. Yang, *et al.*, *Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task*, 2019. arXiv: 1809.08887 `[cs.CL]`.

[254] B. Ahmadnia and R. Aranovich, "An effective optimization method for neural machine translation: The case of english-persian bilingually low-resource scenario," in *Proceedings of the 7th Workshop on Asian Translation*, 2020.

[255] R. Akhoundzade and K. H. Devin, "Unsupervised aspect-based sentiment analysis in the persian language: Extracting and clustering aspects," in *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2020, pp. 94–100.

[256] M. Aliramezani and Doostmohammadi, "Persian sentiment analysis without training data using cross-lingual word embeddings," in *2020 10th International Symposium onTelecommunications (IST)*, IEEE, 2020, pp. 78–82.

[257] S. A. A. Asli and Sabeti, "Optimizing annotation effort using active learning strategies: A sentiment analysis case study in persian," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 2855–2861.

[258] H. Bais and M. Machkour, "French language interface for xml databases," 2020.

[259] F. Balouchzahi and H. Shashirekha, "Puner-parsi ulmfit for named-entity recognition in persian texts," in *Congress on Intelligent Systems*, Springer, 2020, pp. 75–88.

[260] M. E. Basiri and Kabiri, "The effect of aggregation methods on sentiment classification in persian reviews," *Enterprise Information Systems*, vol. 14, no. 9-10, pp. 1394–1421, 2020.

[261] Z. Borhanifard and Basafa, "Persian language understanding in task-oriented dialogue system for online shopping," in *2020 11th International Conference on Information and Knowledge Technology (IKT)*, IEEE, 2020, pp. 79–84.

[262] K. Dashtipour and Gogate, "A hybrid persian sentiment analysis framework: Integrating dependency grammar-based rules and deep neural networks," *Neurocomputing*, vol. 380, pp. 1–10, 2020.

[263] K. Dashtipour and M. Gogate, "A hybrid persian sentiment analysis framework: Integrating dependency grammar-based rules and deep neural networks," *Neurocomputing*, vol. 380, pp. 1–10, 2020.

[264] M. B. Dastgheib and Koleini, "The application of deep learning in persian documents sentiment analysis," *International Journal of Information Science and Management (IJISM)*, vol. 18, no. 1, pp. 1–15, 2020.

[265] M. Dawodi, J. A. Baktash, T. Wada, N. Alam, and M. Z. Joya, "Dari speech classification using deep convolutional neural network," in *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, IEEE, 2020, pp. 1–4.

[266] M. Dawodi, M. Z. Joya, N. Hassanzada, J. A. Baktash, and T. Wada, "A comparative study of machine learning methods and feature extraction methods for dari sentiment analysis," *International Information Institute (Tokyo). Information*, vol. 23, no. 2, pp. 117–137, 2020.

[267] R. Dehkharghani and H. Emami, "A novel approach to sentiment analysis in persian using discourse and external semantic information," *arXiv preprint arXiv:2007.09495*, 2020.

[268] R. Esmailpour and Ebrahimy, "Developing an effective scheme for translation and expansion of persian user queries," *Digital Scholarship in the Humanities*, vol. 35, no. 3, pp. 493–506, 2020.

[269] A.-E. Hassanien, A. T. Azar, T. Gaber, D. Oliva, and F. M. Tolba, Eds., *Proceedings of the international conference on artificial intelligence and computer vision (AICV2020)*. Cham: Springer International Publishing, 2020.

[270] M. Heidari and P. Shamsinejad, "Producing an instagram dataset for persian language sentiment analysis using crowdsourcing method," in *2020 6th International Conference on Web Research (ICWR)*, IEEE, 2020, pp. 284–287.

[271] L. Jafar Tafreshi and F. Soltanzadeh, "A novel approach to conditional random field-based named entity recognition using persian specific features," *Journal of AI and Data Mining*, vol. 8, no. 2, pp. 227–236, 2020.

[272] L. Jafar Tafreshi and F. Soltanzadeh, "A novel approach to conditional random field-based named entity recognition using persian specific features," *Journal of AI and Data Mining*, vol. 8, no. 2, pp. 227–236, 2020.

[273] M. Karrabi and Oskooie, "Sentiment analysis of informal persian texts using embedding informal words and attention-based lstm network," in *2020 8th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, IEEE, 2020, pp. 143–147.

[274] M. E. Khademi and M. Fakhredanesh, "Persian automatic text summarization based on named entity recognition," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, pp. 1–12, 2020.

[275] M. E. Khademi, M. Fakhredanesh, and S. M. Hoseini, "Conceptual persian text summarizer: A new model in continuous vector space," *Int Arab J Inf Technol*, vol. 17, no. 4, pp. 529–538, 2020.

[276] A. Koochari, A. Gharahbagh, and V. Hajihashemi, "A persian part of speech tagging system using the long short-term memory neural network," in *6th Iran. Conf. Signal Process. Intell. Syst. ICSPIS*, vol. 2020, 2020.

[277] S. Lazemi and H. Komleh, "Persian semantic role labeling based on dependency tree," *International Journal of Information Science and Management (IJISM)*, vol. 18, no. 1, pp. 93–108, 2020.

[278] S. Momtazi and F. Torabi, "Named entity recognition in persian text using deep learning," *Signal and Data Processing*, vol. 16, no. 4, pp. 93–112, 2020.

[279] M. Moradshahi, G. Campagna, S. J. Semnani, S. Xu, and M. S. Lam, "Localizing open-ontology qa semantic parsers in a day using machine translation," *arXiv preprint arXiv:2010.05106*, 2020.

[280] A. Oncevay, B. Haddow, and A. Birch, "Bridging linguistic typology and multilingual machine translation with multi-view language representations," *arXiv preprint arXiv:2004.14923*, 2020.

[281] R. A. Pazos-Rangel, R. Florencia-Juarez, M. A. Paredes-Valverde, and G. Rivera, *Handbook of Research on Natural Language Processing and Smart Service Systems*. IGI Global, 2020.

[282] P. Qi, Y. Zhang, and Zhang, "Stanza: A python natural language processing toolkit for many human languages," *arXiv preprint arXiv:2003.07082*, 2020.

[283] M. S. Rasooli and Safari, "The persian dependency treebank made universal," *arXiv preprint arXiv:2009.10205*, 2020.

[284] H. Rouhizadeh and Shamsfard, "Knowledge-based word sense disambiguation with distributional semantic expansion for the persian language," in *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, 2020, pp. 329–335.

[285] M. M. Sadr, A. Mousavi Chelak, S. Ziaei, and J. Tanha, "A predictive model based on machine learning methods to recognize fake persian news on twitter," *International Journal of Nonlinear Analysis and Applications*, vol. 11, pp. 119–128, 2020.

[286] R. Saheb-Nassagh and Asgari, "Repersian: An efficient open information extraction tool in persian," in *2020 6th International Conference on Web Research (ICWR)*, IEEE, 2020, pp. 93–99.

[287] J. P. R. Sharami and Sarabestani, "Deepsentipers: Novel deep learning models trained over proposed augmented persian sentiment corpus," *arXiv preprint arXiv:2004.05328*, 2020.

[288] N. Taghizadeh and Borhanifard, "Nsurl-2019 task 7: Named entity recognition (ner) in farsi," *arXiv preprint arXiv:2003.09029*, 2020.

[289] E. Taher, S. A. Hoseini, and M. Shamsfard, "Beheshti-ner: Persian named entity recognition using bert," *arXiv preprint arXiv:2003.08875*, 2020.

[290] N. Tohidi and C. Dadkhah, "Optimizing the performance of persian multi-objective question answering system," in *16th International Conference on Technical and Physical Problems of Engineering (ICTPE-2020), Istanbul, Turkey*, 2020.

[291] H. Veisi and A. Haji Mani, "Persian speech recognition using deep learning," *International Journal of Speech Technology*, vol. 23, no. 4, pp. 893–905, 2020.

[292] H. Veisi and H. F. Shandi, "A persian medical question answering system," *International Journal on Artificial Intelligence Tools*, vol. 29, no. 06, p. 2 050 019, 2020.

[293] N. Weir, P. Utama, A. Galakatos, *et al.*, “Dbpal: A fully pluggable nl2sql training pipeline,” in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 2347–2361.

[294] N. Abadani and Mozafari, “Parsquad: Persian question answering dataset based on machine translation of squad 2.0,” *International Journal of Web Research*, vol. 4, no. 1, pp. 34–46, 2021.

[295] M. Abbaszade and V. Salari, “Application of quantum natural language processing for language translation,” *IEEE Access*, vol. 9, pp. 130 434–130 448, 2021.

[296] M. Aghajani, A. Badri, and H. Beigy, “Parstwiner: A corpus for named entity recognition in informal persian,” in *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, 2021, pp. 131–136.

[297] B. Ahmadnia and R. Aranovich, “Augmented spanish-persian neural machine translation [augmented spanish-persian neural machine translation],” in *Proceedings of the 13th International Conference on Agents and Artificial Intelligence (ICAART 2021)*, vol. 1, 2021.

[298] B. Ahmadnia, B. J. Dorr, and R. Aranovich, “Impact of filtering generated pseudo bilingual texts in low-resource neural machine translation enhancement: The case of persian-spanish,” *Procedia Computer Science*, vol. 189, pp. 136–141, 2021.

[299] M. Asgari-Bidhendi and Janfada, “Parsner-social: A corpus for named entity recognition in persian social media texts,” *Journal of AI and Data Mining*, vol. 9, no. 2, pp. 181–192, 2021.

[300] M. I. Askari, A. Satariyan, and M. Ranjbar, “Investigating covert and overt errors using machine translation according to house’s (2015) tqa model within academic context,” *Journal of Language and Translation*, vol. 11, no. 3, pp. 57–73, 2021.

[301] S. Besharati and Veisi, “A hybrid statistical and deep learning-based technique for persian part of speech tagging,” *Iran Journal of Computer Science*, vol. 4, no. 1, pp. 35–43, 2021.

[302] U. Brunner and K. Stockinger, “Valuenet: A natural language-to-sql system that learns from database information,” in *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, IEEE, 2021, pp. 2177–2182.

[303] K. Dashtipour and Gogate, “Sentiment analysis of persian movie reviews using deep learning,” *Entropy*, vol. 23, no. 5, p. 596, 2021.

[304] K. Dashtipour and M. Gogate, “A novel context-aware multimodal framework for persian sentiment analysis,” *Neurocomputing*, vol. 457, pp. 377–388, 2021.

[305] K. Dashtipour and M. Gogate, “Adopting transition point technique for persian sentiment analysis,” in *ICOTEN*, 2021.

[306] K. Dashtipour and Ieracitano, “An ensemble-based classification approach for persian sentiment analysis,” in *Progresses in Artificial Intelligence and Neural Systems*, Springer, 2021, pp. 207–215.

[307] A. Dehghani and S. A. Seyyedsalehi, "Time-frequency localization using deep convolutional maxout neural network in persian speech recognition," *arXiv preprint arXiv:2108.03818*, 2021.

[308] A. Ehsani and A. Mousavi, "Designing a persian question answering system based on rhetorical structure theory," *International Journal of Nonlinear Analysis and Applications*, vol. 12, no. Special Issue, pp. 1449–1468, 2021.

[309] R. Etezadi and M. Shamsfard, "A knowledge-based approach for answering complex questions in persian," *arXiv preprint arXiv:2107.02040*, 2021.

[310] H. Fadaei and N. Fatehi Rad, "Culture-specific items: Khaled hosseini's the kite runner in machine and human persian translation," *International Journal of Foreign Language Teaching and Research*, vol. 9, no. 39, pp. 81–90, 2021.

[311] F. J. Farahani and G. Ghassem-Sani, "Bert-persner: A new model for persian named entity recognition," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, 2021, pp. 647–654.

[312] M. Farahani and Gharachorloo, "Parsbert: Transformer-based model for persian language understanding," *Neural Processing Letters*, vol. 53, no. 6, pp. 3831–3847, 2021.

[313] M. Farahani, M. Gharachorloo, and M. Manthouri, "Leveraging parsbert and pretrained mt5 for persian abstractive text summarization," in *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, IEEE, 2021, pp. 1–6.

[314] P. Gholami-Dastgerdi and M.-R. Feizi-Derakhshi, "Part of speech tagging using part of speech sequence graph," *Annals of Data Science*, pp. 1–28, 2021.

[315] M. Heydari and M. Khazeni, "Deep learning-based sentiment analysis in persian language," in *2021 7th International Conference on Web Research (ICWR)*, IEEE, 2021, pp. 287–291.

[316] S. M. Hoseini, "Persian speech emotion recognition approach based on multilayer perceptron," *International Journal of Digital Content Management*, vol. 2, no. 3, pp. 177–187, 2021.

[317] H. Jafarian and Taghavi, "Exploiting bert to improve aspect-based sentiment analysis performance on persian language," in *2021 7th International Conference on Web Research (ICWR)*, IEEE, 2021, pp. 5–8.

[318] N. Jamali and Yaghoobzadeh, "Percqa: Persian community question answering dataset," *arXiv preprint arXiv:2112.13238*, 2021.

[319] S. Kale, V. Gahse, P. Mahatkar, R. Ratnaparkhi, N. Dhawale, and T. Totade, "An intelligent system for transforming natural language queries into sql and its execution," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 1913, 2021, p. 012 146.

[320] A. N. Karimvand and Chegeni, "Sentiment analysis of persian instagram post: A multimodal deep learning approach," in *2021 7th International Conference on Web Research (ICWR)*, IEEE, 2021, pp. 137–141.

[321] R. Khanmohammadi and Mirshafiee, "Prose2poem: The blessing of transformers in translating prose to persian poetry," *arXiv preprint arXiv:2109.14934*, 2021.

[322] D. Khashabi and Cohan, "Parsinlu: A suite of language understanding challenges for persian," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1163–1178, 2021.

[323] M. Mahdi, "Modifying the snlidb (sinhala natural language interface to database)," Ph.D. dissertation, 2021.

[324] P. Mukherjee, A. Chattopadhyay, B. Chakraborty, and D. Nandi, "Natural language query handling using extended knowledge provider system," *International Journal of Knowledge-based and Intelligent Engineering Systems*, vol. 25, no. 1, pp. 1–19, 2021.

[325] P. Mukherjee, A. Chattopadhyay, B. Chakraborty, and D. Nandi, "Natural language query handling using extended knowledge provider system," *International Journal of Knowledge-based and Intelligent Engineering Systems*, vol. 25, no. 1, pp. 1–19, 2021.

[326] R. Oji and S. F. Razavi, "Parsinorm: A persian toolkit for speech processing normalization," in *2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS)*, IEEE, 2021, pp. 1–5.

[327] Y. Pourebrahim, F. Razzazi, and H. Sameti, "Parallel shared hidden layers auto-encoder as a cross-corpus transfer learning approach for unsupervised persian speech emotion recognition," *Signal Processing and Renewable Energy*, vol. 5, no. 4, pp. 83–108, 2021.

[328] S. Rahmani and Fakhrahmad, "Co-occurrence graph-based context adaptation: A new unsupervised approach to word sense disambiguation," *Digital Scholarship in the Humanities*, vol. 36, no. 2, pp. 449–471, 2021.

[329] H. Rouhizadeh and Shamsfard, "Persian semcor: A bag of word sense annotated corpus for the persian language," in *Proceedings of the 11th Global Wordnet Conference*, 2021, pp. 147–156.

[330] H. Rouhizadeh and Shamsfard, "Persian-wsd-corpus: A sense annotated corpus for persian all-words word sense disambiguation," *arXiv preprint arXiv:2107.01540*, 2021.

[331] N. Sabri, A. Edalat, and B. Bahrak, "Sentiment analysis of persian-english code-mixed texts," in *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, IEEE, 2021, pp. 1–4.

[332] M. M. Sadr *et al.*, "The use of lstm neural network to detect fake news on persian twitter," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 11, pp. 6658–6668, 2021.

[333] M. H. Saghayan and Ebrahimi, “Exploring the impact of machine translation on fake news detection: A case study on persian tweets about covid-19,” in *2021 29th Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2021, pp. 540–544.

[334] M. H. Saghayan, S. F. Ebrahimi, and M. Bahrani, “Exploring the impact of machine translation on fake news detection: A case study on persian tweets about covid-19,” in *2021 29th Iranian Conference on Electrical Engineering (ICEE)*, IEEE, 2021, pp. 540–544.

[335] A. Salemi and E. Kebriaei, “Arman: Pre-training with semantically selecting and reordering of sentences for persian abstractive summarization,” *arXiv preprint arXiv:2109.04098*, 2021.

[336] M. Samadi, M. Mousavian, and S. Momtazi, “Persian fake news detection: Neural representation and classification at word and text levels,” *Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 1, pp. 1–11, 2021.

[337] M. Shamsi, M. Divani, and A. Rasouli Kenari, “Designing an avatar-based translator system from persian into persian sign language (psl),” *Technology of Education Journal (TEJ)*, vol. 15, no. 2, pp. 277–290, 2021.

[338] M. Shirghasemi and Bokaei, “The impact of active learning algorithm on a cross-lingual model in a persian sentiment task,” in *2021 7th International Conference on Web Research (ICWR)*, IEEE, 2021, pp. 292–295.

[339] S. Shumaly, M. Yazdinejad, and Y. Guo, “Persian sentiment analysis of an online store independent of pre-processing using convolutional neural network with fasttext embeddings,” *PeerJ Computer Science*, vol. 7, e422, 2021.

[340] H. Singh and A. Oberoi, “Query relational databases in punjabi language,” in *Computational Methods and Data Engineering*, Springer, 2021, pp. 343–357.

[341] H. Singh and A. Oberoi, “Query relational databases in punjabi language,” in *Computational Methods and Data Engineering*, Springer, 2021, pp. 343–357.

[342] C. Tapsai, P. Meesad, and C. Haruechaiyasak, “Natural language interface to database for data retrieval and processing,” *Applied Science and Engineering Progress*, vol. 14, no. 3, pp. 435–446, 2021.

[343] C. Tapsai, P. Meesad, and C. Haruechaiyasak, “Natural language interface to database for data retrieval and processing,” *Applied Science and Engineering Progress*, vol. 14, no. 3, pp. 435–446, 2021.

[344] M. Vazan and J. Razmara, “Jointly modeling aspect and polarity for aspect-based sentiment analysis in persian reviews,” *arXiv preprint arXiv:2109.07680*, 2021.

[345] H. Wu, C. Shen, Z. He, Y. Wang, and X. Xu, “Scada-nli: A natural language query and control interface for distributed systems,” *IEEE Access*, 2021.

[346] A. Yazdani, H. Simchi, and Y. Shekofteh, “Emotion recognition in persian speech using deep neural networks,” in *2021 11th International Conference on Computer Engineering and Knowledge (ICCKE)*, IEEE, 2021, pp. 374–378.

[347] M. M. Abdollah Pour and S. Momtazi, “Comparative study of text representation and learning for persian named entity recognition,” *ETRI Journal*, 2022.

[348] M. Abdous and B. Minaei Bidgoli, “Semantic textual similarity of persian-english sentences using deep learning,” *Journal of Soft Computing and Information Technology*, vol. 11, no. 1, pp. 18–31, 2022.

[349] K. Dashtipour and Gogate, “Extending persian sentiment lexicon with idiomatic expressions for sentiment analysis,” *Social Network Analysis and Mining*, vol. 12, no. 1, pp. 1–13, 2022.

[350] R. Etezadi and Karrabi, “Dadmatools: Natural language processing toolkit for persian language,” in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, 2022, pp. 124–130.

[351] R. Ghasemi and A. Asli, “Deep persian sentiment analysis: Cross-lingual training for low-resource languages,” *Journal of Information Science*, vol. 48, no. 4, pp. 449–462, 2022.

[352] Z. Jahanbakhsh-Nagadeh, M.-R. Feizi-Derakhshi, M. Ramezani, *et al.*, “A model to measure the spread power of rumors,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–25, 2022.

[353] R. Kabiri, S. Karimi, and M. Surdeanu, “Informal persian universal dependency treebank,” *arXiv preprint arXiv:2201.03679*, 2022.

[354] S. Karimi, A. A. Rasel, and M. S. Abdullah, “A review of persian natural language processing: Classical and modern approaches (in progress),” *IEEE*, 2022.

[355] S. Karimi, A. A. Rasel, and M. S. Abdullah, “Natural language query and control interface for database using afghan language,” in *2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, 2022, pp. 1–8. DOI: 10.1109/INISTA55318.2022.9894168.

[356] S. Karimi, A. A. Rasel, and M. S. Abdullah, “Non-english natural language interface to databases: A systematic review (in progress),” 2022.

[357] P. Kavehzadeh and A. Pour, “A transformer-based approach for persian text chunking,” *Journal of AI and Data Mining*, 2022.

[358] J. Mozafari and Kazemi, “Peransel: A novel deep neural network-based system for persian question answering,” *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[359] R. Sadeghi, H. Karbasi, and A. Akbari, “Exappc: A large-scale persian paraphrase detection corpus,” in *2022 8th International Conference on Web Research (ICWR)*, IEEE, 2022, pp. 168–175.

[360] https://www.researchgate.net/publication/317088993_Information_Extraction_with_Speech_Recognition, Accessed: 2021-12-18.

[361] Z. Ansari, F. Almasganj, and Y. Shekofteh, “Eigenspace-based speaker adaptation methods in persian speech recognition systems,”

[362] P. Borkar, “Natural language processing-based approach for hindi language graphical user interface,”

[363] M. Dehghani and E. A. Noughabi, “Sentiment analysis of persian political tweets using machine learning techniques,”

[364] *EasyAsk review*, en, https://reviews.financesonline.com/p/easyask/, Accessed: 2022-1-4.

[365] A. Fatullayev and N. Sadighian, “Stages of the development of persian-azerbaijani mt dictionary,”

[366] I. E. Gámez, M. de los Ángeles Marrujo, and O. P. García, “Translation of spanish statistics expressions to sql,” *Advances in Soft Computing Algorithms*, p. 155,

[367] *GNOME do - release*, https://do.cooperteam.net/release.shtml, Accessed: 2022-8-26.

[368] J. Kapoor and H. Kaur, “Punjabi natural language interface to database,”

[369] S. Kokabpaik and M. Fathi, “A survey on the satisfaction of efl learners towards google machine translation (english to persian, persian to english) and its accuracy as perceived,”

[370] F. Mohammadpour, A. A. Ahangar, and N. Jahangiri, “Building a hybrid machine translation system for translating from english into persian,”

[371] O. M. Nasab and Mozhgani, “A special domain question answering system for persian,” *Phoenix-company NLP research lab, Narmak, Tehran, Iran*,

[372] *Natural language interface accessibility user requirements*, en, https://www.w3.org/TR/naur/, Accessed: 2022-8-17.

[373] *Natural language interface accessibility user requirements*, en, https://www.w3.org/TR/naur/, Accessed: 2022-8-7.

[374] *Natural language interface accessibility user requirements*, en, https://www.w3.org/TR/naur/, Accessed: 2022-8-26.

[375] *Siri*, en, https://www.apple.com/siri/, Accessed: 2022-8-26.

[376] *So much for a quiet launch*, en, https://blog.wolframalpha.com/2009/05/08/so-much-for-a-quiet-launch/, Accessed: 2022-8-26.

[377] M. Vasheghani Farahani *et al.*, “Adequacy in machine vs. human translation: A comparative study of english and persian languages,” *Applied Linguistics Research Journal*, vol. 4, no. 5, pp. 84–104,

[378] *Why study persian (farsi)?* en, https://www.bu.edu/wll/home/why-study-persian/., Accessed: 2022-8-17.

Additional Thesis Documents

inista 2022

IEEE SMC Systems, Man, and Cybernetics Society

The 16th International Conference on INnovations in Intelligent SysTems and Applications (INISTA'22)

August 8-10, 2022
Biarritz, France

CERTIFICATE OF PARTICIPATION

We hereby are please to certify that

Sadullah Karimi

has attended the 16th International Conference on INnovations in Intelligent SysTems and Applications to present her/his paper

Natural Language Query and Control Interface for Database Using Afghan Language

Co-authored by:
Sadullah Karimi, Annajiat Alim Rasel and Matin Saad Abdullah

AhmedKH

AHMED KHEMIRI

INISTA'22 Organizing Chair

OpenCEMS
Connected Environment & Distributed Data Management Solutions

IEEE COPYRIGHT AND CONSENT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

Natural Language Query and Control Interface for Database Using Afghan Language

Sadullah Karimi, Annajiat Alim Rasel, Matin Saad Abdullah

2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the IEEE PSPB Operations Manual.
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

You have indicated that you DO wish to have video/audio recordings made of your conference presentation under terms and conditions set forth in "Consent and Release."

CONSENT AND RELEASE

1. ln the event the author makes a presentation based upon the Work at a conference hosted or sponsored in whole or in part by the IEEE, the author, in consideration for his/her participation in the conference, hereby grants the IEEE the unlimited, worldwide, irrevocable permission to use, distribute, publish, license, exhibit, record, digitize, broadcast, reproduce and archive, in any format or medium, whether now known or hereafter developed: (a) his/her presentation and comments at the conference; (b) any written materials or multimedia files used in connection with his/her presentation; and (c) any recorded interviews of him/her (collectively, the "Presentation"). The permission granted includes the transcription and reproduction of the Presentation for inclusion in products sold or distributed by IEEE and live or recorded broadcast of the Presentation during or after the conference.
2. In connection with the permission granted in Section 1, the author hereby grants IEEE the unlimited, worldwide, irrevocable right to use his/her name, picture, likeness, voice and biographical information as part of the advertisement, distribution and sale of products incorporating the Work or Presentation, and releases IEEE from any claim based on right of privacy or publicity.

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Sadullah Karimi	07-07-2022
Signature	**Date (dd-mm-yyyy)**

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use.The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from that funding agency.

AUTHOR ONLINE USE

- **Personal Servers**. Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the