

AVCL: Audio Video clustering for learning Conversation labeling using Neural Network and NLP

by

Salman Mostafiz Chowdhury
17101149

Ali Ahammed Rohid
17101361

Rizwan Hussain
17301092

Chowdhury Sujana Mostafa
21101107

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

Department of Computer Science and Engineering
Brac University
January 2022

© 2022. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing 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. We have acknowledged all main sources of help.

Student's Full Name & Signature:



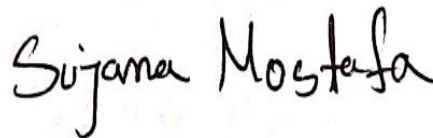
Salman Mostafiz Chowdhury
17101149



Ali Ahammed Rohid
17101361



Rizwan Hussain
17301092



Chowdhury Sujana Mostafa
21101107

Approval

The thesis titled “AVCL: Audiovisual Speech Recognition for Learning Conversational Labeling using Neural Networks and NLP.” submitted by

1. Salman Mostafiz Chowdhury (17101149)
2. Ali Ahammed Rohid (17101361)
3. Rizwan Hussain (17301092)
4. Chowdhury Sujana Mostafa (21101107)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 18, 2022.

Examining Committee:

Supervisor:
(Member)



Moin Mostakim
Senior Lecturer
Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Computer Science and Engineering
BRAC University

Head of Department:
(Member)

Sadia Hamid Kazi
Associate Professor
Computer Science and Engineering
BRAC University

Abstract

Audiovisual data is the most extensively used and abundantly distributed type of data on the internet in today's information and communication age. However, the necessary audiovisual data is challenging to retrieve because the majority of them are not correctly categorized. As a result, it is difficult to locate the necessary audiovisual data in times of need, and as a result, a great amount of potentially useful information does not reach users in a timely manner. A piece of data is only as good as the time frame in which it was acquired. Additionally, because audiovisual files such as lecture notes, recorded classes, and recorded conversations are quite large, skimming through a large amount of audiovisual data for the necessary information can be time consuming, and the likelihood of not receiving the appropriate information on time is relatively high. As a result, we propose a novel model that will take any audio or video file as input and label it according to its content utilizing Convolutional Neural Networks, BERT, and several machine learning techniques. Our proposed model accepts any audiovisual file as input and extracts features from the contents and uses convolutional neural network and transformer to recognize and transcript the speeches of the conversations. Using BERT models and cosine similarity keywords and phrases are extracted from the transcript and the input file will be labeled with the key phrases and keywords that are most similar to the context of the content. Finally, the input file will be appropriately labeled with these key phrases and keywords so that anyone in the future in need of similar information can quickly locate this audiovisual file.

Keywords: Speech Recognition; wav2vec2; BERT; AVSpeech; Keywords extraction; Neural Networks; NLP

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.

Secondly, to our supervisor Mr. Moin Mostakim sir for his kind support and advice in our work. He helped us whenever we needed help.

Thirdly, to all the researchers before us who prospered machine learning field, all the reviews they gave helped us a lot in our later works.

And finally to our parents without their throughout support it may not be possible. With their kind support and prayer we are now on the verge of our graduation.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Acknowledgment	iv
Table of Contents	v
List of Figures	vii
List of Tables	viii
1 Introduction	1
2 Background And Related Work	5
2.1 Speech Recognition	5
2.2 NLP : Contextualized Keyword Extraction	7
2.3 Audiovisual Quality Enhancement	9
2.4 Other Related Works	11
3 Methods	14
3.1 Speech recognition	14
3.1.1 Wav2vec2	14
3.2 NLP : Keyword Extraction	16
3.2.1 BERT	16
3.2.2 KeyBERT	17
3.3 AVCL Model	18
4 Datasets Descriptions	19

4.1	LibriSpeech	19
4.2	AVSpeech	19
4.3	Preprocessing for evaluation	20
4.4	Train Test Amount	20
4.5	Audio Video Datasets	20
5	Experiments	23
5.1	Fine tuning wav2vec2	23
5.2	Evaluation in Low Resources	23
5.3	Punctuation and Capitalization	24
5.4	Measuring Word Error Rate(WER) and Character Error Rate(CER)	24
5.5	Keyword Extraction N-gram Model	25
5.6	Max Sum Similarity	26
5.7	Maximal Marginal Relevance	26
5.8	Determining Centroid for clustering	26
6	Results	28
7	Contribution, Challenges, Future work	35
7.1	Our contribution	35
7.2	Challenges	35
7.3	Future work	36
8	Conclusion	38
	Bibliography	43

List of Figures

1.1	Importance of Audiovisual Data	1
3.1	wav2vec2 Model	15
3.2	wav2vec2 Model 2	15
3.3	BERT model	16
3.4	KeyBERT Model	17
3.5	AVCL Model	18
5.1	Memory Error figure	24
5.2	Punctuation 1	24
5.3	Punctuation 2	24
5.4	Scatter Plot : WER	25
5.5	Scatter Plot : CER	25
5.6	keyword Extraction Example	26
6.1	Result Scatter Plot : WER	30
6.2	Bar Plot : WER	30
6.3	Bar Plot : CER	31
6.4	Bar Plot : WER and CER	32

List of Tables

4.1	Librispeech Training set amount	20
4.2	Librispeech Test set amount	20
4.3	Audiovisual Datasets table	21
4.4	Audiovisual Datasets table	22
6.1	WER value of different wav2vec2 models. We compare our model AVCL with other wav2vec2 model in terms of WER value.	28
6.2	CER value of different wav2vec2 models. We compare our model AVCL with other wav2vec2 model in terms of CER value.	29
6.3	WER value of different models. We compare our model AVCL with speech recogniton model in terms of WER value.	29
6.4	Extracted keywords for five randomly chosen audiovisual file.	33
6.5	Extracted keywords with relevance of randomly chosen audiovisual file.	33

Chapter 1

Introduction

In this twenty-first century, audiovisual data has risen to become one of the most essential types of information available around the world, and audiovisual data is now required for practically all parts of our lives, whether they be educational, business, or daily living [1] activities. As a result, it is critical for us to ensure the right use of audiovisual information for the advancement of our society's development. Large amounts of critical information can be lost or misplaced if they are not properly preserved or made accessible and discoverable. However, because the significance of these data has not been assessed on a large scale from the beginning, we are bound to find many discrepancies in them, and for that reason, even after having a large amount of extremely important and valuable data, we are still in desperate need of audiovisual data. Undoubtedly, one of the most significant reasons for these inconsistencies is that the audiovisual data has not been properly labeled; in fact, many of these data have not been labeled at all in certain instances.

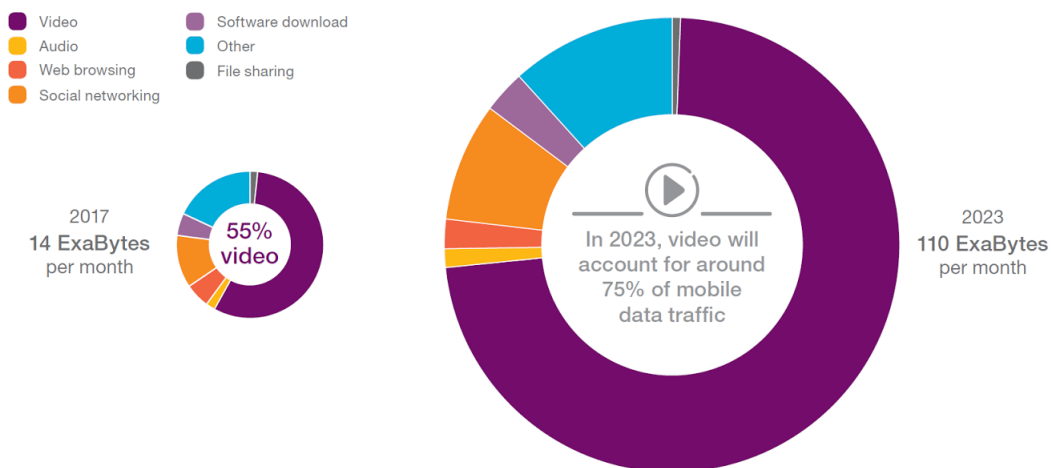


Figure 1.1: Importance of audiovisual data in today's world. Almost 55 % of the total data over the internet is audiovisual and by 2023, the amount will rise around 75 % . ¹

¹<https://www.ericsson.com/en/reports-and-papers/consumerlab/reports/tv-and-media-2017>

More importantly, without sufficient labeling, it can be time-consuming and hard for anybody to identify the materials they need during an emergency situation. For example, during this epidemic, institutions from all over the globe have gone online to communicate with one another. There seems to be a limitless number of video lectures being created for this purpose on the internet, and they are all easily accessible. Schools, colleges, and universities, for example, have been delivering their lessons and lectures through various online platforms, resulting in the creation of over a thousand lecture Videos during the pandemic. The majority of the videos, on the other hand, are not correctly labeled; rather, they are labeled with the code of the platform that was utilized during those online lectures, which is confusing. All around the globe, audiovisual data is playing an important role not only during this epidemic but also throughout the process of learning and teaching in many disciplines. All of this audiovisual content is quite valuable and may be used in a variety of areas of learning and teaching [2]. There is a plethora of knowledge included inside this vast collection of video lectures; yet, owing to a lack of clear labeling, it might be difficult to locate the material you are looking for within this massive collection of videos. This sort of material must thus be properly labeled in order to be used to its full potential.

Clustering is a traditional data mining, information retrieval, and knowledge discovery technique based on machine learning that divides abstract objects into similar objects and finds hidden patterns from data of different classes. Clustering splits data into several subsets where every cluster consists of data objects with high inter-correlation and low intra-correlation. However, semantic connotations of those terms are not taken into consideration while clustering documents. For this reason, semantic document clustering, which takes meaningful associations into account, can achieve higher accuracy, and for this, more informative variables such as concept weight are particularly important. Though document clustering has been studied for decades, it is still far from a trivial solution. The significant hurdles lie in selecting the appropriate features from the documents that need to be clustered, finding proper similarity among documents, using suitable clustering methods and efficient and optimized algorithms. The problem faced in semantic document clustering which results in poor results is because of not considering the semantics relationship in the document. Furthermore, because of the synonym and polysemous problems in the conversations, it is impossible to designate the precise content of a document and no meaningful grouping can take place at this time. Polysemy and synonymy are two of the most common issues with unsupervised learning methods. As a result, document clustering approaches that take semantic meaning into account are an absolute necessity.

Audio-video clustering is the method of grouping audio video files together based on criteria that are similar between the items in the group. One of the objectives is to organize groups of audio visual data that have similar traits together into clusters. Due to the fact that the mixed multiple object, as well as its sounds, can be interpreted easily, audio video datas is intractable [3], making it easy and efficient to match the unconstrained environment. However, the modalities of audio and video in any audiovisual data are very closely connected and related, the information contained in these two modalities is completely different, and thus can be used to extract multiple important informations from a single segment of audiovisual data.

Taking advantage of this high degree of correlation and similarity, semantics can be retrieved from audio video files with greater precision when both audio and video modalities are utilized [4]. In recent years, academics from all over the world have been studying clustering algorithms because of their extensive utility and significance [5][6]. With the importance of clustering comes the fact that the algorithms are simple to grasp and are widely used [7], and therefore audio-video clustering has become one of the most fascinating study subjects throughout the world for better understanding and applications of audiovisual data.

Labeling audiovisual conversation depends on recognising the speeches in the audiovisual file, without proper extraction of the speakers' speeches; the semantic meaning of the content cannot be labeled. So in order to extract the exact speeches from the audiovisual file different aspects need to be taken into account such as the background noise in the audio-video samples, there may be speakers of the same gender whose voices are indistinguishable [8], or different types of conversations may overlap. Again, there may be difficulties in accurately grouping semantics and labeling conversations.

Another important consideration while labeling audio video conversations is the number of speakers speaking as well as the gender of speakers. Speakers from the same gender might be a trouble as they often share the same characteristics [9]. So there is a high probability that while speakers of the same gender are speaking their speeches might overlap. Therefore some important parts of the conversation might not be recognized. Much earlier speech separation approaches exhibit reduced performance when attempting separate speech mixtures comprising same-gender speech [10]. In most parties, multiple people are talking, and there are other background noises involved, so separating those voices individually and labeling is a bit complicated. The technical difficulty is to automatically identify people in a discussion while also capturing the silences and pauses they can create [11]. Failing to identify the speeches properly as well as partially identified speeches can cause severe problems while labeling as the context of the conversation can change due to these phenomena.

For correct labeling of audio video conversations, the conversations need to be converted into text first [12] and then semantics will be applied on the text to extract meaning and keywords from the text document. So a proper conversation into a text document must be ensured because if the whole conversation is not properly converted into text, the meaning and keywords from the text will not align with the topic of the audio video file and the meaning of the material cannot be comprehended. As a result the labeling will be incorrect. One crucial consideration in the conversion of voice to text is the dialect of the speakers, which can be extremely difficult to deal with. As a result, it can be extremely difficult to classify the content.

When it comes down to it, audio speech recognition (ASR) is a useful link between these two components: humans and machines. When it comes to noisy data, machines are less accurate and perform worse than they should. Humans, on the other hand, perceive speech in a multimodal manner [13]. This model is useful in voice recognition since it scans the complete input sequence before anticipating the output sentence, which makes it faster and more accurate. It is incredibly difficult to distinguish spoken words in a noisy environment. It is more difficult to make mistakes while listening to audio only classification rather than listening to audio and video

classification since the visual serves as a reference point to compare to, for example, when lip reading.

In this work, we aim to develop an architecture for audio video conversation labeling that will aid in the labeling of any audio video file according to its content, which will be useful when searching for the necessary information. Furthermore, our work will aid in the classification of any unnamed audiovisual files and the clustering of those files according to their label, allowing related audiovisual files to be identified in the same location. We hope to design a model that will accept any audio-visual data as input and, as a final output, will classify the themes of a file according to the substance of the data. It is possible to extract features from a given input by employing deep neural networks. The audio and video are combined in order to increase the precision and accuracy of the model, as well as to improve the quality of voice separation. Speeches are transformed into text files, and then, using Natural Language Processing, the necessary labels from the input are retrieved and used to generate conversation labeling for audio and video data. We aim to contribute in efficient searching of audiovisual content, organizing unorganized audiovisual files and clustering the similar ones into the same group, precisely labeling any unlabeled and unnamed audiovisual files.

Chapter 2

Background And Related Work

2.1 Speech Recognition

Takaki et al.[14] in their work presented a recurrent neural network(RNN) based architecture for large scale audio video speech recognition. To be more specific their architecture includes a recurrent neural network transducer(RNN-T), which is a sequence-to-sequence model.Makino, Takaki, et al construct their large dataset from youtube building upon the semi supervised method taken into [15] and taking the approach of [16]. For their work, they used YTDEV18 and a generally available LRS3-TED set. While synchronizing the audio video features they sample sized the audios in single channel, 16kHz sample and as for the video features, the final frame rate of 33 fps was taken into consideration. By illustrating the reasons behind the unsynchronization between audio and video features, they down-sampled the videos into 25fps which eventually helped for synchronizing the audio and video features. STFT was used in order to analyze the audios against the video features at 25 ms rate in proportion to the video features. These synchronized audios were then fed into a four hundred dimensional stacked encoder and the video features were fed into a 5 layer convolutional neural network. As both of these worked as audio and video switches correspondingly, these were passed into the next encoder of BiLSTM. Both encoder and decoder then merged into a joint layer and from their softmax was applied to get output. This architecture shows improved performance on audio only systems even in noise and overlapping speech and compared to the YTDEV18 dataset the LRS3-TED dataset shows more improved results.

In today's world, creating speech is a highly classified process, but what if the data is noisy, During the training phase, if the audio is denoised and features are retrieved, the performance is exceptional, Devton et al. [17] conducted two sets of noise handling tests in their research. To begin with, larger models can be used without affecting the bitrate in the input. Second, the signal-to-noise ratio is different (SNR). Working with noisy data can be difficult since it contains both clean and interfering signals. They proposed the ConvTasNet denoiser in this study, which samples data audios at 16KHz with a window size of 4ms and a step size of 1. There were 256 filters in each of the filter banks. According to them, denoiser produced the best results.

In the field of audiovisual speech recognition, a voice recognition system developed

by Noda et al.[11] is considered to be one of the most promising solutions available currently (AVSR). This work provides an overview of the HMM system for AVSRs that are noise-resistant. According to the result of the research, denoised MFCCs may be able to increase word recognition rates by around 65 percent when the SNR of the audio signal input is 10dB or higher. Deep neural networks (DNNs) have lately gotten a lot of interest in the machine learning community because of their ability to extract significant features from data that may be used by a variety of voice recognition methods. The proposed models were developed using data from a Japanese audiovisual dataset. The dataset consists of 400 words (216 phonologically phrased words) and 184 key phrases extracted from the ATR speech corpus, which have been integrated to produce the final product. In order to generate the acoustic model, twenty four thousands phrase samples (one set of phrases per participant; approximately one hour of dialogue in total) were created. An area of the mouth was manually taken out of the raw data and reduced in size to 16 pixels, which was then utilized for training and evaluating visual models of the mouth. Effective matching in an infinite environment is difficult due to the large number of different things and noises that must be considered.

In this work, Cheng et al. [18] introduced a novel end-to-end model called w2v-cifbert, which includes wav2vec2.0 (pre-trained audio dataset), BERT (pre-trained text corpus), and a CIF method to connect the two modules above. Pre-trained ASR models were combined with a pre-trained linguistic encoder (BERT), creating an end-to-end ASR model that was able to recognize speech. During finetuning on minimal labeled data, the combined model just learns speech to language transformation. A monotonic attention technique with no additional parameters matches the length of the two modalities. An entirely new layer is provided for the concealed mapping across modalities, as well. In addition, we propose scheduling fine-tuning of the pre-trained linguistic encoder’s capacity to model text context. Experiments demonstrate that pre-trained modules can be used effectively. This model outperforms previous end-to-end models in terms of recognition performance on the CALLHOME corpus (15 hours). They concentrated on low-cost ASR and tested primarily with the CALLHOME corpus. For each language, there are less than 20 hours of transcriptions in the CALLHOME corpus. The transcription is processed using the free source wav2vec2 encoder and the bert-base linguistic encoder for English. Fine adjustments are made to the wav2vec2 encoder and Adam optimizer is employed. A random-initialized decoder is to blame for the poor performance of w2v-seq2seq models, even with cold-fusion, according to the researchers. This means that compared to cold-fusion, their strategy can effectively utilize pre-trained LMs and achieve an improved outcome.

In their study Khurana et al.[19] proposed a straightforward model for speech recognition using wav2vec2. They showed that wav2vec2 works really well in a monolingual environment for speech recognition for many languages. In their paper, they improved the performance of wav2vec2 with the use of Dropout Uncertainty-Driven Self-Training, otherwise known as DUST. One of the major achievements of their study is finding that the wav2vec2 works as best as the XLSR model.

Kessler et al. [20] in their study used wav2vec2 in a self-supervised environment on multiple languages for speech recognition. The authors proposed a speech representation method for multiple languages. For this, they used continual learning tech-

niques in a self-supervised environment. The authors explained that self-supervised representations and fine-tuning on a datasets is an optimistic approach in the field of speech recognition as there is a multitude of unannotated speeches. The proposed wav2vec2 model can successfully reduce the training time of a new language without forgetting previously trained and learned languages.

Yi et al. [21] also in their study made use of wav2vec2 for speech recognition and achieved quite better performance compared to previous works. The authors mentioned the prominent ability of wav2vec2 for speech representation in a very low resource speech recognition environment. In this paper, the authors verify the credibility of wav2vec2 in speech recognition in other languages along with English and achieved better performance in six other languages.

2.2 NLP : Contextualized Keyword Extraction

In their study,Chahine et al. [22] proposed a system that as an output gives contextualized keywords of any given documents in order to index them according to their context. Their study aimed to provide an automatic indexing system by extracting the meaningful and contextual information as keywords. The authors divided the information extraction into two parts; extraction from the text and about the text. For the first part dictionary-based models [23], the rule-based models [24], the probabilistic models [25], Conditional Random Fields [26] are used to extract concepts, named entities, relations between entities and concepts. The second activity consists in extracting information to describe the text as a whole and the same methods can be used for this purpose as well. This study also mentioned latent methods and for this three correlation of corpus can be used; words and documents, words and words, lastly documents and documents. Finally Abi Chahine, Carlo, et al are categorized and clustered by rule-based CONTRUE system [27] or the machine learning Naive Bayes Classifiers (NBC) [28] and Support Vector Machines (SVM) [29].

Different people speak in a variety of dialects, and comprehension is the most crucial component of communication. Humans convey their emotions through words, but just because people speak the same language does not mean that they are easy to understand. Every person on the planet has their own unique rhythm. Translation from speech to text, followed by summarization, according to Vinnarasu & Jose [30] is a significant aspect of the modern period, as they discuss in their work. Speech extraction appears to be relatively simple, but it is considerably more sophisticated than it appears. The fact that the computer needs to train in multiple languages means that there are differences in pronunciation and, last but not least, pauses in the phrases, among other things. Text summary is quite popular among this generation because most people are unable to read long texts and become bored, and it takes a significant amount of time. In this study, the authors address the use of Natural Language Processing (NLP) to extract features from spoken language. They used the Google API because it is free and open source, but the difficulty is that the text is not in sentence style, which makes it difficult to use (missing.) Python String Tokenizer is a standard package that is used to separate the phrases from one another in Python. Despite the fact that much work has been done in the areas of text summarization and speech recognition, there is still much more

work to be done in this area. The Gensim library was utilized for the purpose of text summarization. Topic identification is accomplished by the use of word vectors. Because punctuation is included in the provided text, it takes less time to summarize and categorize the topic.

Shukla & Kakkar, in their study [31] emphasised that, the most critical step when working with text data is keyword extraction. Extracting keywords enables the reader to focus on the most significant parts of the text rather than reading the entire text. This article discusses a technique for extracting keywords from educational video transcripts from MOOC's that use a Regular Expression Grammar Rule approach. The technique is based on identifying Syntactic Bits inside the transcript's content. Extracting keywords assists in identifying the instructional material's most critical sections. The suggested approach is designed to extract keywords (Syntactic Phrase) from internet video lecture transcripts. The dataset used in this research is a collection of freely accessible recordings from NPTEL courses. Transcripts are available in Adobe Acrobat (PDF) format. A PDF to XML translator is used to retrieve information such as text (transcripts of educational videos), the course title, and the lecturer's name. For ease of handling and execution, this metadata data is held in an XML file. The target is to extract phrases from the textual data in such a way that the extracted keywords characterize both the phrase and the topic mentioned in the text. NP Tag patterns (Grammer) procedures are used to successfully extract the keywords along with their PoS and Chunk Tags. This method is proven to be straightforward and robust in its ability to extract terms and phrases from text since it allows the user to specify the type of tag sequences to compare. Nevertheless, a limitation of this grammar-based strategy is that it is hard to develop a system of regulations that captures all NP fragments and no verb due to the complexity of the Noun Phrase structure.

Utilizing a BiLSTM-CRF, Sahrawat et al.[32] showed a model for extracting keyphrases from academic publications using deep contextualized embeddings as a sequence labeling task. BERT and SciBERT's self-attention layers, as well as their predictions, were examined in detail in a case study. Using contextual embeddings and domain-specific language models may aid improve keyphrase extraction in other NLP applications, according to the researchers. For the purpose of keyphrase extraction, they used a BiLSTM-CRF to solve a labeling job, and then used several contextual embedding architectures to represent the underlying words. This sequence labeling problem was solved using a BiLSTM-CRF architecture, where each token in the input text was mapped to a fixed-size dense vector. When it came time to encode the order in which the word representations appeared, researchers turned to the BiLSTM. An affine transformation is also utilized to translate the BiLSTM output to the class space afterwards. The BiLSTM score outputs are fed into a CRF layer as input. An output label sequence's likelihood is calculated in a CRF by multiplying the scores by a constant and then averaging over all potential output label sequences. The final model outperforms all of the other SemEval-2010 models. Upon further examination of the data, it was shown that SciBERT performed better than other models in identifying keyphrases containing scientific terminology.

In their paper Vaswani et al. [33] a simple network architecture that utilizes attention mechanisms and transformers along with recurrence and convolutions and achieves better performance. This model is known as BERT. The experiments done

on the papers show better performance while taking very little time to train.

Tang et al.[34] in their study make use of the BERT model to extract keywords to classify progress notes. Even though there are various models for keywords extraction, keywords extraction from clinical notes is really challenging. The authors researched the attention-based learning models to extract keywords from clinical progress notes from a real-life EHR system to classify them. Their study shows significant accuracy when the BERT model is fine-tuned with attention layers.

Lim et al.[35] fine-tuned the BERT model for extracting keywords in scientific articles which achieves good results. The authors explained the challenges in keyword extraction from critical data of model times and worked on improving this by using the BERT model. In their study, the used BERT along with SciBERT is a BERT model trained on scientific data. For their work, the authors used three different datasets containing scientific data and achieved better performance in keyword extraction.

2.3 Audiovisual Quality Enhancement

Malm et al.[36] worked on improving contrast and reducing noises from videos which were taken on a very low level light and presented a system to improve the overall quality of the videos. In order to preserve and enhance blurred images and fine tuning spatial details the authors used smoothing kernels. The proposed system is very easy to use and only a handful of predefined features have been used, this system can show improved results if done using a GPU. This model is inspired by [37] and from this the authors have done their modifications and improved its performance and applicability to make it suitable for low light lever vision objectives. The results are very efficient and show quite an improvement in reducing noise and enhancing contrast of low light videos.

The use of facial movement aids in speech recognition. Using a single channel to extract the target speaker is discussed in detail in this study by Morrone et al.[38] and others. Visual features are being mapped using the Time Frequency (TF) technique. Acoustic missing speech spectrograms were developed later as a result of this. The authors also state that they use two different types of datasets. TCD-Timid and GRID are two words that come to mind. The single channel method is the most effective, and the results are rather beautiful. Pre-trained Convolutional Neural Networks were used to generate clean spectrograms for non-audible video, and they were the most effective of the neural networks tested. CNN has been taught in such a way that it can evaluate flawless speeches and video input that is provided. Using the mobility of the entire face is far more effective for tracking and conducting the experiment. The majority of the time, the target is facing the camera, but what if the situation is more complicated? Three separate audio mixed samples were prepared, and the ADAM optimizer was used to train the model on each of the samples. All of the audios were captured at 16KHz, and a Short Time Fourier (STFT) algorithm was applied with a HANN window of 25 samples for a total of 400 samples in each audio. The FFT was of size 512 with a hop length of 10ms, and it was employed. The power law was calculated using the value $p=0.3$ as a starting point. Additionally, they used mean per speaker 0 and 1 standard deviation

normalization per speaker. When using several speakers, it is preferable to use a single channel rather than many channels. Motion vectors based on landmarks were employed for TF. The video was sampled from 24 frames per second to 100 frames per second in order to match the audio spectrograms frames per second.

In their study Chuang et al.[39] proposed the goal of combining visual data as an emergency backup and supplementary contribution to reduce the noise level of loud auditory data; over a hundred studies have proved the impact of AVSE tasks. Researchers proposed the LAVSE algorithm to improve audio-visual speech efficacy. The LAVSE system uses less internet arithmetic than standard AVSE systems and only partially addresses user privacy and security concerns. These improvements were made by embracing three common practical concerns when adopting AVSE structures: supplemental visual data, audio-visual asynchronization, and bad video data. In order to improve LAVSE, the suggested method uses convolutional recurrent neural networks instead of the traditional linear AVSE model. The researchers found iLAVSE efficient on the Taiwan Mandarin speech with visual dataset. The simulation results show that iLAVSE can accurately overcome the three primary difficulties described above as well as improve enhancement performance. The findings confirm that iLAVSE is useful in situations where high audio-visual detectors are not always available. iLAVSE contains three stages: pre-processing, AVSE using CRNN, and data restoration. The iLAVSE system has three steps. The preprocessing step uses a CRQ module and an AE module in conjunction with an AE module to extract a compact latent representation for the AVSE stage. In the studies, feeding the AVSE stage only 0.33 percent ($1/307.2 \times 100\%$) of the initial visual reduced the intake size, resulting in improved PESQ and STOI ratings. We were able to fix the audiovisual asynchronization and low-quality visual data issues using data augmentation and zero-pad training. In the experiments, iLAVSE outperformed AOSE and comparable AVSE systems, proving that it can overcome the three practical hurdles and outperform AOSE. As a result, the findings verified that the proposed iLAVSE system is robust and can be used in real-world applications. The researchers hope to expand the proposed model’s neural network designs, objective functions, and compression techniques in the future. The TMSV dataset is presently being created into a real-world analogue. The real-world dataset is expected to include audio-visual data collected with various devices in diverse contexts. We will then use the newly obtained data to evaluate the SE systems. We believe we can improve our systems’ robustness in future conditions including impulse responses, frame rates, and noisy surroundings. Meanwhile, we’re working on improving the iLAVSE system’s usability by combining visual information with self-supervised and meta learning.

Giri et al.[40] presented the Attention Wave-U-Net structure for speech improvement in this study, which allows the model to focus on the most important aspects of the raw waveform. They presented a U-Net architecture with a local self-attention mechanism for the purpose of speech improvement. Using the VCTK dataset’s speech quality measurements, they found that the attention mechanism greatly improved denoising quality, beating all other published speech enhancement methods on this dataset to reach state-of-the-art values. A detector of low-level vocal activity can be inferred from the final-layer attention mask, the researchers found. Using attention approaches for speech enhancement has never been done before, according to this study, according to the authors. By multiplying the down-block feature with

an attention mask, they can find significant features in their suggested design instead of concatenating features computed along the contracting path with the same hierarchical level among the up blocks. The CSTR VCTK Corpus, comprised of 56 speakers - 28 men and 28 women - with a variety of accents, was utilized to test the new technique. It was shown that their proposed voice denoising/enhancement strategy could be effective as part of an acoustic front-end for an Automatic Speech Recognition (ASR) system, in addition to increasing perceptual speech quality measures. They employed a free ASR system that has been trained to recognize conversational speech. A total of four conditions were provided for WER: The baseline model (Wave-UNet) and their proposed technique with attention and augmentation process the clean and the noisy data, respectively. When compared to a baseline model, the proposed approach has a WER improvement that is greater than the baseline model. Due to the harder conditions, the performance disparity is larger. At long last, preliminary acoustic front end results for an ASR system showed that this model has the potential to be a very valuable component.

2.4 Other Related Works

For speech separation, Lu et al.[41] on their study, aims to cluster individual voices from audio or video input of multiple simultaneous talkers where the CASA, HMM, and NMF were used for the audios and for audio-visual methods, different methods like Noise-Invariant model, Look to Listen, AV-Match method, Sound-of-Pixel were used. In this paper, for clustering of audio networks BiLSTM and for visual networks VGG-style CNN were applied. The proposed AVDC model of this paper predicts time frequency masks for the speakers which are later used to build source signals with the help of spectrograms. As for the dataset for the model GIRD and TCD_TIMIT dataset were used and both of these datasets contain audiovisual data. In processing of data, audio files were processed as e downsampled to 8 kHz to compute the STFT, and to direct the mouth whereas from the visual data Python implementation- Coarse2Fine algorithm was used. Moreover, for audio-only baseline the DC model and uPIT model, and for audiovisual baseline the AC-Match model was used. From the experiment robust improvement of the performance of this model for the partially missing visual information was displayed.

If the visual features are present, it is possible to improve the quality of speech separation. In their article, Ephrat et al.[10] discussed how they used the open-source Google API to create a three-second clip at twenty frames per second (fps) that contained 75 face thumbnails, which was then used for video representation. Short Time Fourier (STFT) is a technique that is primarily and commonly used in applications such as noise reduction and other similar tasks. In this experiment, three seconds audio segments were extracted from the audio characteristics and used to test the hypothesis. Power Law compression was employed to keep louder audio from overpowering the weaker audio that came before it. For the purpose of learning audio and video features, a Dilated Convolution Neural Network (CNN) was deployed. This experiment employed a batch size of six samples and trained it with the ADAM optimizer for five million steps at a learning rate of 0.00002999. The use of 16KHz sampling for audio was also acknowledged by the researchers.

Again, mono was used for the sound rather than stereo, as was the case previously. In order to compute STFT, they used a hop length of 10ms, a HANN window size of 25ms, and an FFT with a size of 512 to accomplish so. Finally, with $p=0.3$, Power Law compression was used to compress the data. For training the model, on the other hand, they had to create a new audio video dataset that included visible speakers and clearer audio.

Due to the variety of mixed items and noises in an endless setting, effective matching is impossible. To overcome this issue, the authors of the article, Hu et al.[3], advised that individuals learn how to effectively absorb auditory and visual components. DMC is a ground-breaking unsupervised audiovisual learning technique that simultaneously performs several clustering operations by recording separate audiovisual correspondences utilizing multimodal vectors of convolutional modeling in various shared domains. After developing the DMC system, the collinear networks should have been taught to respond to a variety of stimuli or acoustic conditions. As a result, the article advises evaluating the internal CNN layers' learned auditory and visual representations. To enhance efficiency, the DMC model was trained on 0.4 million randomly picked unclassified movies from the SoundNet Flickr dataset. The audio and visual messages are identical to those in, except that each movie is constructed from pairs of five-second sound clips and non-overlapping graphics. Notably, the 1.6 million audiovisual pairings comprised are roughly 17 times fewer than in L3 and five times fewer than in SoundNet. Additionally, end-to-end training is achievable for an integrated multimodal clustering network with max-margin degradation. The ESC-50 file contains 2000 audio samples, each lasting five seconds. The feature maps are pooled to extract the audio representations. The mean accuracy of the five leave-one-fold-out assessments is determined. This level of performance is achieved by utilizing only 0.4 million training data points (just .4 million video data). By exploiting more audiovisual interactions in the unrestricted films, the model can efficiently train the unimodal network. Human accuracy is 0.813 on this dataset. The PASCAL VOC 2007 dataset used in this investigation contains twenty item types collected in real-world scenarios. Global pooling over the VGG16 network was used to acquire the visual attributes. Additionally, the classifier is a cross one-vs-all optimized SVM. To facilitate comparison, the best convolution/pooling characteristics of several techniques are chosen. Instead of just learning the audio and visual representations sequentially, as described previously, the DMC approach simultaneously learns the visual and auditory interpretations. The purpose of this paper is to investigate the complex association between audio and visual information in an unconstrained environment using the proposed deep multimodal clustering algorithm. In comparison to the previous approximation, the model can be trained to have more effective auditory and visual features more optimally, surpassing humans. Additionally, this level of sophistication yields significant advances in difficult audiovisual tasks such as sound specificity, multi - source SED, and audiovisual comprehension. Numerous trials in feature evaluation and audiovisual employment are done. The findings indicate that DMC is capable of forming a more effective unimodal representation than humans. Additionally, DMC excels at sound localization, multisource recognition, and audiovisual comprehension.

In the paper [42], the author MacQueen studied different methods of classification and analysis where the author studied k-means clustering and mentioned that the k-

mean clustering algorithm is theoretically very prominent. The author in his paper discussed the concept of the k-means algorithm where he mentioned that it is a generalization of an ordinary sample mean with pertinent asymptotic behavior and as for the object of the k-mean algorithm is to classify sample data into k numbers of group.

Chapter 3

Methods

We have separated our entire work into two sections: one is concerned with detecting speech from audiovisual input and producing text as an output, and the other is concerned with extracting contextualized keywords from text and labeling the content. In this section, we have given a brief overview of the methodology we used for our research.

3.1 Speech recognition

First, we have given an overview of our methodology for speech recognition. For this purpose there are many methods that can be used but for our study, we used wav2vec2.

3.1.1 Wav2vec2

This model, which has been pretrained, receives audio in its raw form as an input and produces a representation of talks based on the timestamp of the audio file on the other end. Wav2vec2 has a multilayer convolutional neural networks encoder that takes the wave as an input and encodes it. The output of the encoder is then passed on to the next layer, which is a Transformer, which eventually builds representations for capturing the entirety of the information contained in the supplied sequence [43][44][45]. Specifically, the objectives are represented by this quantization module in the self-supervised objective. To capture the interdependence of the complete representation, self-attention has been employed in conjunction with context representation, which has been constructed in this model.

With regard to the first encoder, the raw waveform is fed into a number of layers of temporal convolution neural networks, each of which is connected to a layer that performs normalizations [46] upto zero mean and unit variance. As for the activation function, it has been implemented using the GELU [47]. The number of timestamps is decided based on the sum of strides utilized in the temporal convolution encoder, and the output is then passed as an input to be used by the Transformer. This layer consists of a context network along with Transformers architecture [33][43][48]. In this layer relative positional embedding is used. In the next part the output from

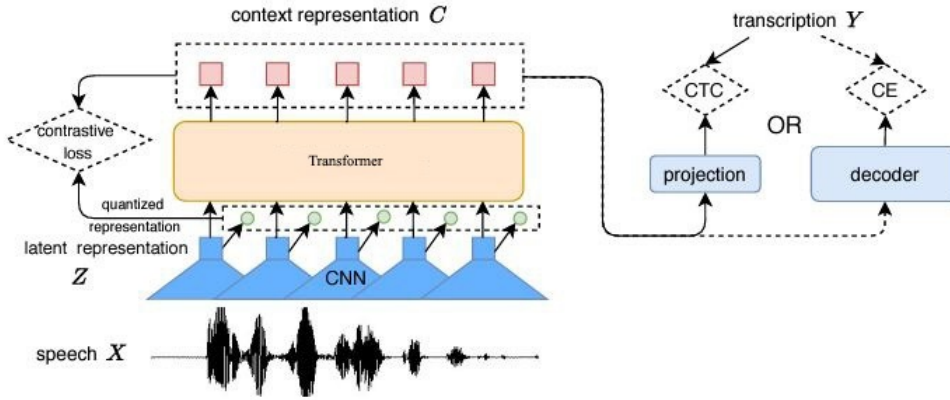


Figure 3.1: Left: the structure of wav2vec2.0 and corresponding self-training criterion. It contains a stack of convolution layers and self-attention layers; Right: two decoding branches that apply wav2vec2.0 to ASR tasks with additional projection or decoder, which is trained with CTC or cross-entropy loss respectively

the encoder is discretized to a finite number of speech representations and for this task product quantizations[49] have been applied. Here, Gumbel Softmax has been used which differentially selects discrete codebooks.

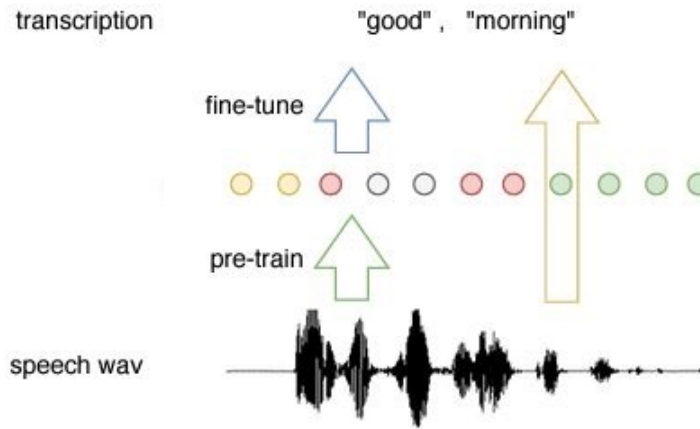


Figure 3.2: Demonstration of how wav2vec2.0 pre-trained. The model is pushed to learn distinguishable fine-grained acoustic units during pre-training phase. They will merge into coarser-grained ones to fit the target ASR task during the finetuning phase.

This version of wav2vec2 has been pre trained in a similar fashion to language model where in the latent feature encoder an amount of time steps has been masked. This has been fed to the context network where a feature vector has been shared among all the masked time which has been trained. In the pre-trained version, the speech recognition has been done from contrastive tasks.

$$L = L_m + \alpha L_d \tag{3.1}$$

Where α is hyperparameter in 3.1.

The contrastive loss has been determined as following :

$$L_m = -\log\left(\frac{\exp(\text{sim}(c_t, q_t)/\kappa)}{\sum_{\bar{q}_t \sim Q_t} \exp(\text{sim}(c_t, \bar{q}_t)/\kappa)}\right) \quad (3.2)$$

where sim is cosine similarity in 3.2

3.2 NLP : Keyword Extraction

In this section, we have given an overview of our language model that we used for extracting contextualized keywords from the transcript of the previous section. For this purpose we used BERT and KeyBERT.

3.2.1 BERT

BERT, which stands for Bidirectional Encoder Representations from Transformer[33], is a pre-trained model used in Natural Language Processing tasks, it uses surrounding text to establish context from the text which eventually helps to understand the meaning of language of that text which is normally ambiguous. Pre-training and fine-tuning are the two steps that need to be done while implementing BERT. At first, for fine-tuning parameters which are pre-trained has to be initialized, but if the fine-tuning is done with the same parameters twice there will emerge two separate and new models.

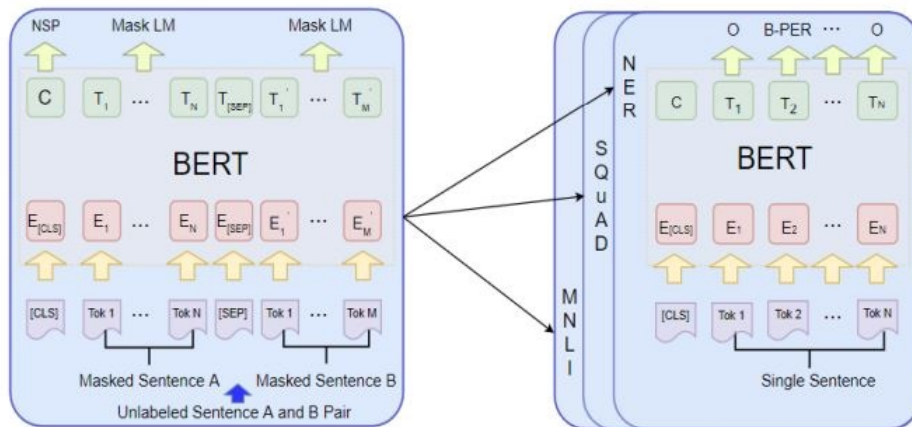


Figure 3.3: Two steps in the implementation of BERT: (Left) Pre-training and (Right) Fine-tuning

There is a unified architecture for BERT's feature that is capable of doing different tasks and as for BERT architecture model, there are multiple layers of bidirectional transformer encoder. BERT came with two models, Base and Large. The Base

model consists of 12 layers of 768 hidden size with 110M parameters and 12 self-attention heads. Whereas, the large model consists of 24 layers of 1024 hidden size, 16 self-attention heads, and 340M total parameters[43]. There are basically two steps of BERT, pre-training and tuning the model for respected tasks. For specific tasks, 11 new NLP tasks were performed using the model for fine-tuning, which achieved better performances. Nonetheless, when it comes to contextualized key phrases or keywords extraction, BERT architecture can be very useful and can show better performance.

3.2.2 KeyBERT

KeyBERT[50] is a BERT based keyword extraction technique which leverages BERT embeddings and cosine similarity to find the keywords and keyphrases from text which are mostly similar with the context to the text itself. KeyBERT at first extract the document embeddings from the given input which give a document level representation.

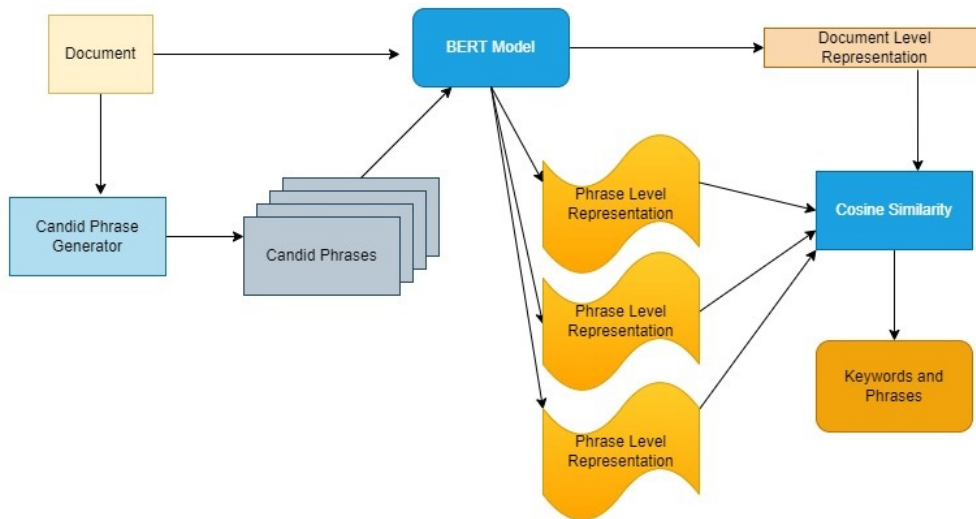


Figure 3.4: KeyBERT model : Generates Keywords using BERT.

Then from the representation word embeddings are extracted for the N-gram words and phrases. To find the most similar words and keyphrases from the texts which are closest to the context of the text cosine similarity has been used. There are more than twelve embedding models included in keyBERT and can be used easily. Each of the embedding models provide higher results. For our work, we chose the ‘all-MiniLM-L12-v2’ embedding model which was originally proposed in [51]. All the embedding models including our chosen one shows impressive performance in keywords extraction.

3.3 AVCL Model

In this section, we present the detail view of our architecture where for speech recognition we used wav2vec2 and for NLP keyword extraction we applied keyBERT.

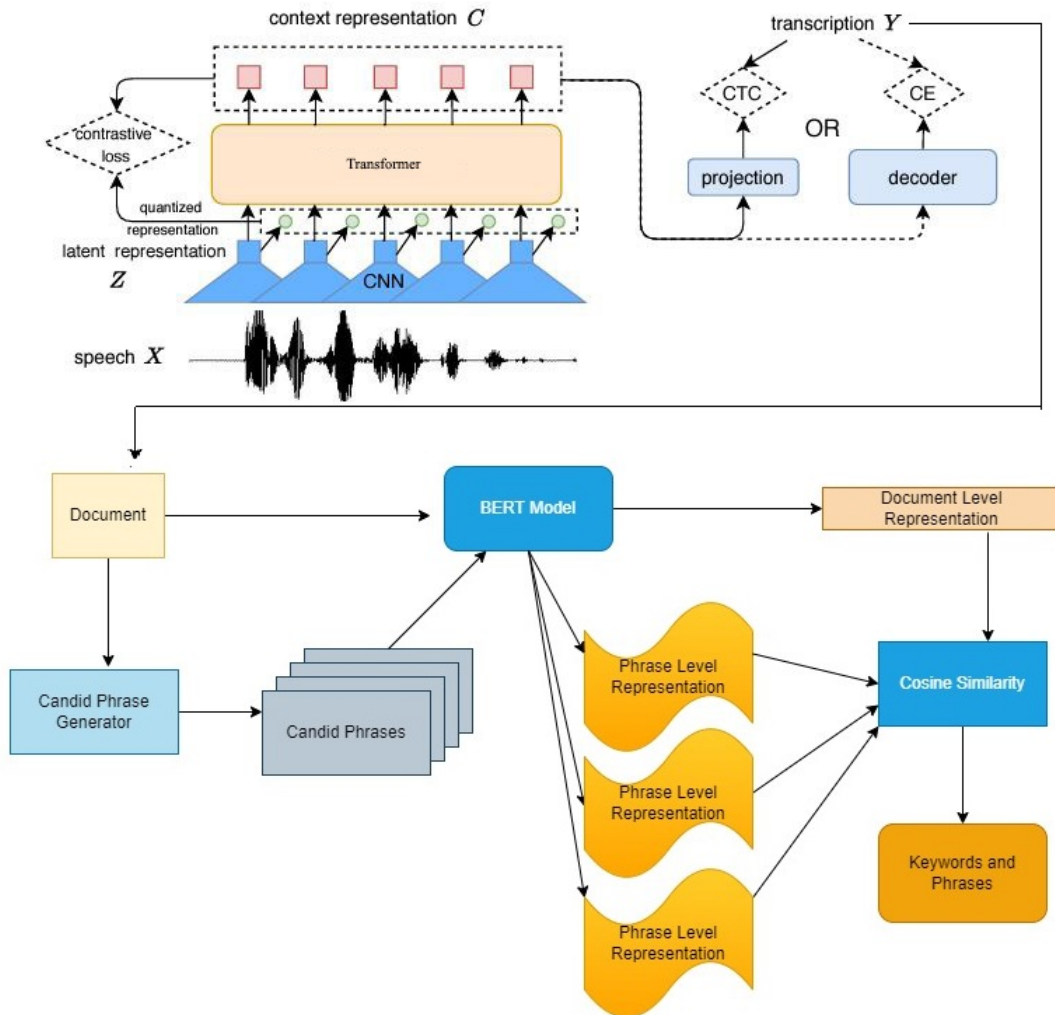


Figure 3.5: AVCL model: This is a two part model, first part is speech recognition where wav2vec2 is used and the second part is NLP where keyBERT is used for keywords extraction.

Chapter 4

Datasets Descriptions

In this part of the paper, we have given an overview of all the datasets that we used in our study and presented a table of Audiovisual datasets with aspiration of helping the future researches in the same field.

4.1 LibriSpeech

For our work, we applied the wav2vec2 model that had been pre-trained with LibriSpeech[52], and we are using this pre-trained model to recognize speech. The dataset includes approximately 1000 hours of 16KHZ English voice recording at different locations. The LibriVox project has collected data on Librispeech from audio books that have been made available online. Librispeech is a solid choice for tasks involving speech recognition. There are two types of settings in this dataset: clean and noisy. Clean audio had a lower Word Error Rate (wer) and vice versa, whereas noisy audio had a higher Word Error Rate (wer). For training purposes, the dataset was divided into two sections, one of which had 100 hours of speech and the other of which contained 360 hours of speech. This model was pre-trained on 64 V100 GPUs for about two days utilizing 64 V100 GPUs.

4.2 AVSpeech

For evaluating our model, we used the dataset created by Ephrat et al.[10] 2018 called AVSpeech. AVSpeech is nonetheless a dynamic new media dataset of more than 1TB which contains audio-visual speech video clips which are mostly 3-10 seconds clips. AVSpeech contains three hundred thousand videos comprising more than 4000+ hours of video clips of different lectures, talk shows, and tutorial videos in different broad spectrum of people, dialects, and nonverbal cues from YouTube which is beneficial for training the model and can be downloaded using youtube-dl and can be framed using FFmpeg. Each clip has a unique voice auditory clean sounds, along with a single talking figure well evidently seen in the film. Considering all these features, AVSpeech is the ideal dataset for our research purpose.

4.3 Preprocessing for evaluation

We have used AVspeech dataset for evaluating the whole model. For the AVSpeech dataset, we have used YouTube-dl to download the dataset. The range of the video is set to 0 to 20 for this purpose. After that, we framed the video to our desired measure. We extract 75 frames corresponding to each video clip using 25FPS using FFmpeg. As mentioned before AVSpeech dataset contains more than 4500 hours of videos but we did not extract all those hours of videos rather we only extracted around 3000 hours of clips from the dataset. Following this, we downloaded the audios for each data in the range of 0 to 20, the same as video data. Sox was used for audio conversion and the FFmpeg library for handling the audio and video files. All the audio files are then resampled into 16kHz.

4.4 Train Test Amount

The train test amount we have used to fine-tune the wav2vec2 is given at table 4.1 and 4.2

Table 4.1: Librispeech Training set amount

Subset	Hours	Per-speaker minutes	Female speaker	Male speaker	Total speaker
train-10h	10	25	12	12	24
train-1h	1	2.5	12	12	24
train-10min	10min	2.5	2	2	4

Table 4.2: Librispeech Test set amount

Subset	Hours	Per-speaker minutes	Female speaker	Male speaker	Total speaker
train-clean	5.4	8	20	20	40
train-other	5.1	10	17	16	33

4.5 Audio Video Datasets

In this section we present tables 4.3 and 4.4 of audiovisual datasets that we came across during our study. As these datasets are scattered all over the place, we tried to provide an accumulation of audiovisual datasets with hope of helping future researchers in this field.

Table 4.3: Audiovisual Datasets table

Dataset Name	Description
GIRD[53]	This dataset consists of high-quality audiovisual recordings of 1000 sentences spoken by 34 speakers.
Kinetics[54]	This dataset is a high-quality, large-scale human activity recognition dataset. The dataset contains around 500,000 video clips covering 600 human action classes (600 clips per action class). Each video clip is roughly 10 seconds long and has only one action class.
AudioSet[55]	AudioSet is a collection of nearly 2 million human-annotated 10-second video snippets. These clips are from YouTube, thus many are low-quality and include many sound sources. These data are annotated using a 632 event ontology.
VoxCeleb1[56]	This audio-visual dataset extracts short samples of human speech from YouTube interview films. Voices of Celebrities (VoxCeleb) It contains around 1280 speakers and over 200k samples.
ESC-50[57]	The ESC dataset is a consistent collection of brief environmental recordings (5-second-long clips, 44.1 kHz, single channel). The ESC-50 dataset contains 2000 5s audio segments.
Flickr-SoundNet[58]	It has over 2 million videos, however we pick a random subset of 500k (400k training, 50k validation and 50k test).
WHAM! [59]	WHAM! is a noisy dataset that couples each two-speaker mixing in the wsj0-2mix dataset with an individual random noise background scene. The noise recordings were made in late Restaurants, cafes, bars, and parks dominate the settings.
TCD-TIMIT[60]	TCD-TIMIT is another audiovisual speech dataset commonly used in audio-visual speech separation research.
ATR speech database[61]	Six males' ATR voice data were used in the dataset (400 words: 216 phonetically balanced words and 184 significant words). A total of 24000 words were recorded (one set of words per speaker; approximately 1 hours of speech in total).

Table 4.4: Audiovisual Datasets table

Dataset Name	Description
LRS3-TED[62]	LRS3-TED is a multi-modal speech recognition dataset. It features over 400 hours of TED and TEDx films, subtitles, and word alignment restrictions.
MovieQA[63]	Video and text story comprehension using the MovieQA dataset. The data set contains about 15,000 multiple choice answers from over 400 films, with considerable semantic diversity.
LRS2[13]	Thousands of spoken words from BBC broadcast. Each sentence is 100 characters. Date of broadcast divides training, validation, and test sets.
Million Song Dataset[64]	This dataset is a collection that is free to use which consists of audio characteristics and information for a million songs from today’s popular music. The Echo Nest provided feature analysis and information for one million songs, which served as the collection’s foundation. There is no audio in the collection; only the computed attributes are included.
Sentibank [65]	Nearly 500,000 photos, each identified with one of 1,533 adjective-noun pairings. Emotional concepts are pairs like “charming house” and “ugly fish”. The SentiWordnet lexicon maps each concept to a sentiment value.

Chapter 5

Experiments

In this chapter we have discussed about the experiments we carried out in our study.

5.1 Fine tuning wav2vec2

The model we chose for our work is pre-trained and we fine-tune the model on labeled data. On top of the Transformer, an output layer is added which is generated randomly to predict characters. For all subsets, in two different learning rates we train for fine-tuning. The learning rates are 0.00002 and 0.00003 respectively. We have chosen the configuration which provides the lower Word Error Rate. Learning rate is heuristically tuned until fine-tuning has become stable. For the Base version, a learning rate of 0.0001 is chosen. The learning rate increased for the first 1/10th of the updates, then for the 4/10th of the updates it was constant and for the rest it slowly decreased, here, Adam optimizer and a tri-state rate scheduling are used. Batch size of the Base version per GPU was higher than the Large version but while fine tuning, the Large version needed more GPU. For fine-tuning, only the output classifier and the transformer are updated. By fine-tuning the wav2vec2 model, it achieves a lower WER than before.

5.2 Evaluation in Low Resources

We evaluate different lengths of audiovisual files in our pre-trained model and record the WER values against each file. The short videos worked pretty well as the model could take the whole file and recognize the speech together and give output. But when it comes to longer videos, a higher GPU is needed to take the whole conversation as input. When we tried to evaluate our model for longer videos with lower GPU and faced `RunTimeError`.

To overcome this, we chunk the file into parts. We use a module of Python called `pydub`¹. It is a Python library that only works in audio files. With this library audio files can be splitted, merged and edited. We use `audiosegment` class of `pydub` to chunk the audio files. We chunk the files based on the silences of the audio files

¹<https://github.com/jiaaro/pydub>


```
RuntimeError: CUDA out of memory. Tried to allocate 1.21 GiB (GPU 0; 11.17 GiB total capacity; 9.78 GiB already allocated; 785.81 MiB free; 9.80 GiB reserved in total by PyTorch) If reserved memory is >> allocated memory try setting max_split_size_mb to avoid fragmentation. See documentation for Memory Management and PYTORCH_CUDA_ALLOC_CONF
```

Figure 5.1: Memory error: Longer videos need more GPUs, without sufficient memory it shows RunTimeError.

when there was a minimum of 500ms of silence in between the conversations, we chunk there. For this we use `split_on_silence` of the `pydub.silence` class. All these chunks are saved together in order and separately recognized and transcribed. After all the chunks are transcribed, we concatenate them together and give us the whole transcription of the conversation.

5.3 Punctuation and Capitalization

For better understanding and representation of the whole conversation we use punctuation and capitalization. We use a toolkit built for working with NLP in python called NLTK². It provides various text processing libraries and with this tokenizing, parse tree visualization etc can be done. Punkt³ Sentence tokenizer is used to divide the whole conversation into a list of sentences. Punkt uses an unsupervised algorithm to build a model for abbreviation words, collocations and words that start sentences. The NLTK data package includes a pre-trained Punkt tokenizer for English.

```
Representing intages in pythonFirstly a quick refresher what is an intergerAn
```

Figure 5.2: Without punctuation words from different sentences get merged.

```
Representing intages in python. Firstly a quick refresher what is an interger.
```

Figure 5.3: With punctuation proper representation of sentences is achieved.

Without punctuation, the WER value increases quite a lot where, when punctuation is used WER value decreases substantially.

5.4 Measuring Word Error Rate(WER) and Character Error Rate(CER)

To evaluate our model, we measured the word error rate(WER) and character error rate(CER) for the audiovisual datas and for this we used a python package called JiWER⁴. Using a speech-to-text transcription, it calculates the minimum edit distance between the ground truth and the hypothesis sentences.

²<https://www.nltk.org/>

³<https://www.kite.com/python/docs/nltk.punkt>

⁴<https://pypi.org/project/jiwer/>

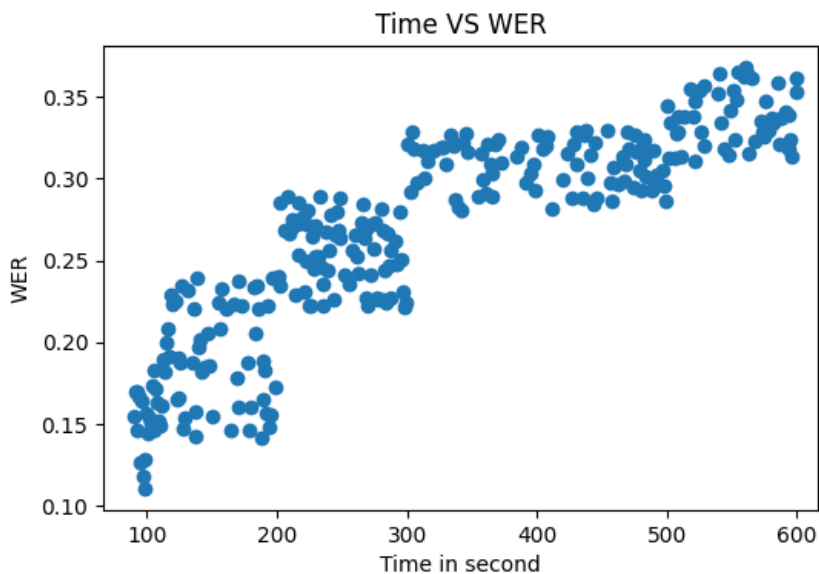


Figure 5.4: Demonstration of WER value while speech recognition with time. WER values increases with time but not always. In this figure WER value up to 10minutes(600 seconds) are shown.

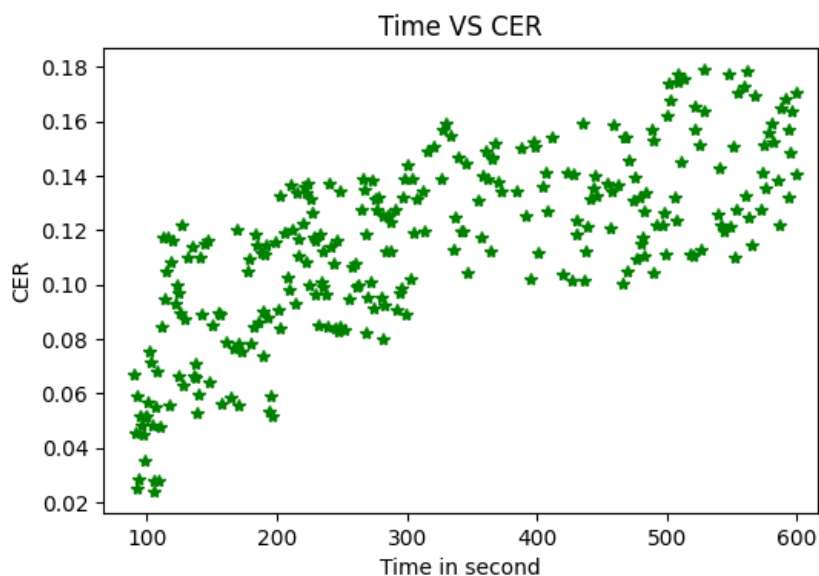


Figure 5.5: Demonstration of CER value while speech recognition with time. CER values increases with time but not always. In this figure CER value up to 10minutes(600 seconds) are shown.

5.5 Keyword Extraction N-gram Model

In the transcription of speech recognition, we use N-gram language model and cosine similarity to extract keywords and phrases which are most similar to the context of the speech. N-gram is a contiguous sequence of n items from a given sample of text or speech in the fields of computational linguistics and probability. The items can

be phonemes, syllables, letters, words or base pairs according to the application. The n-grams typically are collected from a text or speech corpus.

How to move an object on a photo using Adobe Photoshop. First of all open the **image** on which you want to make changes. Once done double click on it in the layers panel to convert it to layer 0 from **background** layer. After this make a selection around the object which needs to be moved using the polygonal **lasso** tool from the toolbar on the left. Now press ctrl X to cut the selection area and create a new layer from layers panel. Press ctrl V to paste the object on the new layer. Adjust the position of the object and use the eraser tool to rub off its edges so that it merges with the background. Next go back to layer 0 there will be a **patch** missing from the image. Make a selection around the **patch** using the polygonal **lasso** tool. Now go to the Edit menu at the top and select fill from this list of options. When the dialog box appears set the use to content-aware mode to normal opacity to 100% and press ok. **Photoshop** will now automatically fill the missing patch. If the **patch** is too large apply this option in 2 or 3 parts. You can then use the smudge tool to even out the area. In the end select both layers and press ctrl e to merge them.

Figure 5.6: Demonstration of how wav2vec2.0 pre-trained. The model is pushed to learn distinguishable fine-grained acoustic units during pre-training phase. They will merge into coarser-grained ones to fit the target ASR task during the finetuning phase.

We set the range of used n-gram models from one upto three which gives us one, two or three keywords or phrases based on the cosine similarity of the words. Depending on the cosine similarity the best keywords are selected to label the content.

$$similarity(A, B) = \frac{A \cdot B}{\|A \cdot B\|} = \frac{\sum_{i=1}^n A_i * B_i}{\sqrt{\sum_{i=1}^n A_i^2} * \sqrt{\sum_{i=1}^n B_i^2}} \quad (5.1)$$

5.6 Max Sum Similarity

For diversifying our results on keywords extraction, we take 2 x **top_n** the most similar keywords and phrases to the document. After the keywords are selected, we take all the **top_n** combinations from the selected 2 x **top_n** words and extract the combinations that are least similar to each other. To decide the similarity here, we use cosine similarity.

5.7 Maximal Marginal Relevance

Along with Max sum similarity, we used Maximal Marginal Relevance(MMR) to widen our results on keywords and key phrases extraction. Maximal Marginal Relevance is introduced in [46], it tries to reduce the redundancy of results while at the same time maintaining query relevance of results for already ranked documents/phrases. We have used cosine similarity here to determine the most similar words and phrases against the document.

5.8 Determining Centroid for clustering

After keywords extraction, we label the audiovisual files based on the most relevant keywords that have been extracted. On the label we perform k-mean clustering to

group the audiovisual files. For clustering we need to determine the correct number of clusters or centroids on which the cluster will be performed. For this we calculated the sum of squared errors for every k in k-means clustering.

$$SSE_{1..k} = \sum_{i=1}^k \sum_{x \in C_i} dist^2(m_i, x) \quad (5.2)$$

Here, x= data point in cluster C_i and m_i is representative point of the cluster in 5.2

All the sum of square error values added up to get the actual sum of square error values. Then using the elbow method we chose the number of k in k-means clustering.

$$SSE = SSE_1 + SSE_2 + SSE_3 + SSE_4 + \dots + SSE_k \quad (5.3)$$

Chapter 6

Results

In this part of the paper, we have discussed the results acquired from our research.

The results of our experiment have been discussed in detail in this section. Our pre-trained model is compared to other models that have been utilized for speech recognition in the following table 6.1 and 6.2. Specifically, we examine the word error rate (WER) and the character error rate (CER) of the models. On the basis of the table, it can be concluded that wav2vec2-large-960-lv60 produces reasonably decent results in terms of WER. This version has a WER of over 24%, which is remarkable. However, when the CER value is taken into consideration, wav2vac2-xlsr-en produces improved results. With a CER of 10.75 percent, it was a success. This is due to the fact that this model has already been pre-trained to take punctuation into account. As a result, it produces a superior outcome in CER tests. Other models, such as wav2vec2-large-960h, wav2vec2-base-960h, and wav2vec2-base-100h, had a WER of 33 percent, 40 percent, and 51 percent, respectively, compared to the baseline model. As for CER, the wav2vec2-large-960h had a lower CER (16%) than the other models in the comparison. The CER of the other two models is 19 percent and 25 percent, respectively. Wav2vac2 has extremely high WER and CER values after being trained on the timid dataset for speech recognition.

Table 6.1: WER value of different wav2vec2 models. We compare our model AVCL with other wav2vec2 model in terms of WER value.

Model Name	WER%
AVCL(wav2vec2-large-960h)	21.03%
wav2vec2-large-960h-lv60	23.97
xlsr-en-punctuation	29.10
wav2vec2-large-960h	32.79
wav2vec2-base-960h	39.86
wav2vec2-base-100h	51.06
wav2vec2-large-lv60-timit-asr	59.96
wav2vec2-base-10k-voxpuli-ft-en	66.41
wav2vec2-base-timit-asr	68.78

Table 6.2: CER value of different wav2vec2 models. We compare our model AVCL with other wav2vec2 model in terms of CER value.

Model Name	CER%
AVCL(wav2vec2-large-960h)	10.39
wav2vec2-large-960h-lv60	11.14
xlsr-en-punctuation	10.75
wav2vec2-large-960h	16.03
wav2vec2-base-960h	19.89
wav2vec2-base-100h	25.06
wav2vec2-large-lv60-timit-asr	34.28
wav2vec2-base-10k-voxpopuli-ft-en	36.76
wav2vec2-base-timit-asr	36.81

Wav2vec2-large-lv60-timit-asr has a WER of about 60% and a CER of 34%, whereas wav2vec2-base-timit-asr has a WER and a CER of 69 percent and 37 percent, respectively, whereas In our work, we implemented a pre-trained model that outperforms all of the other models stated above, including the one that was used in our previous research. With a WER of 21 percent and a CER value of only 10 percent, our model outperformed the competition. Our model surpassed the other model in part because we fine-tuned our pre-trained model, which provided us with an exceptional WER value in the first instance. By fine-tuning our pre-trained model, we were able to generate WER values that were nearly 4-5 percent better, resulting in a WER value of 21 percent for our model. On the other hand, we take into consideration the punctuation of language in the same way as xlsr-en-punctuation does. As a result, our model produces outcomes that are almost a little bit better than the competition. Our model achieves a CER of 10% in this instance.

In 6.3, we compare our model (AVCL) to other models in terms of WER, and the results are presented in the table. We can see from the table that mlstm-residual has a WER of 43.93, which is a good result. Also included is a comparison of our model to other models such as Speech-Transformer, Iterative Pseudo-labeling(IPL). With a WER of 26.02, IPL outperforms all of the other models, while the Speech-Transformer has a WER of 33.77, making it the worst performer of the bunch.

Table 6.3: WER value of different models. We compare our model AVCL with speech recognition model in terms of WER value.

Model Name	WER%
mlstm-residual[66]	43.93
Speech-Transformer[67]	33.77
Iter. pseudo-labeling[68]	26.02
AVCL	21.03%

Finally, the mlstm-residual is 43.93. Our model outperforms all of the other models in terms of performance. The WER rate achieved by AVCL is 21.03 percent.

In fig 6.1, we show a scatter plot of our evaluation of WER. From the scatter plot it can be seen that, with time the WER value increases but it hardly goes above the

WER of .35. Comparing the amount of data in the range of .30 to .35, the amount of data below .30 is much greater in number.

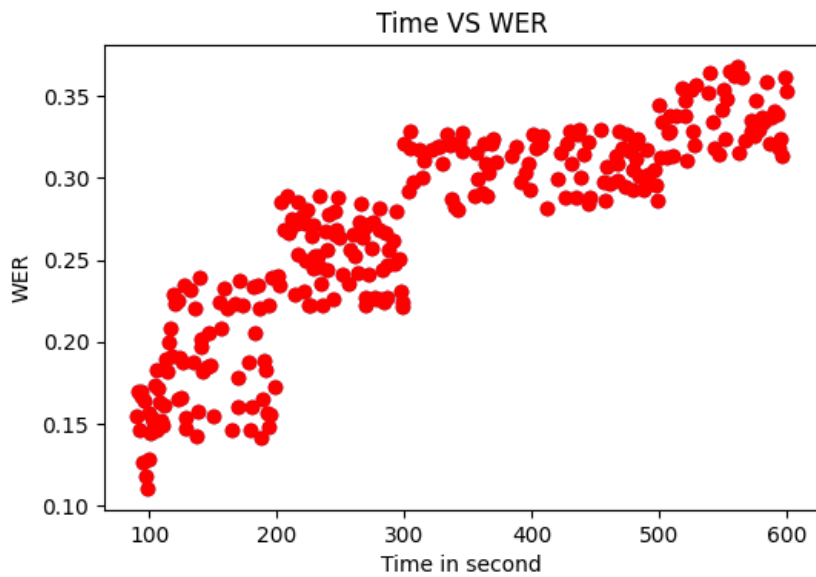


Figure 6.1: WER scatter plot of AVCL in speech recognition.

In figure 6.2 we present graphical view of average WER calculated for each hundred seconds. From one hundred seconds up to six hundred seconds for each hundred seconds we calculated the average WER values. It shows the WER values depends on time of the data but not that much. Rather it more rely upon the clarity of words in speech recognition.

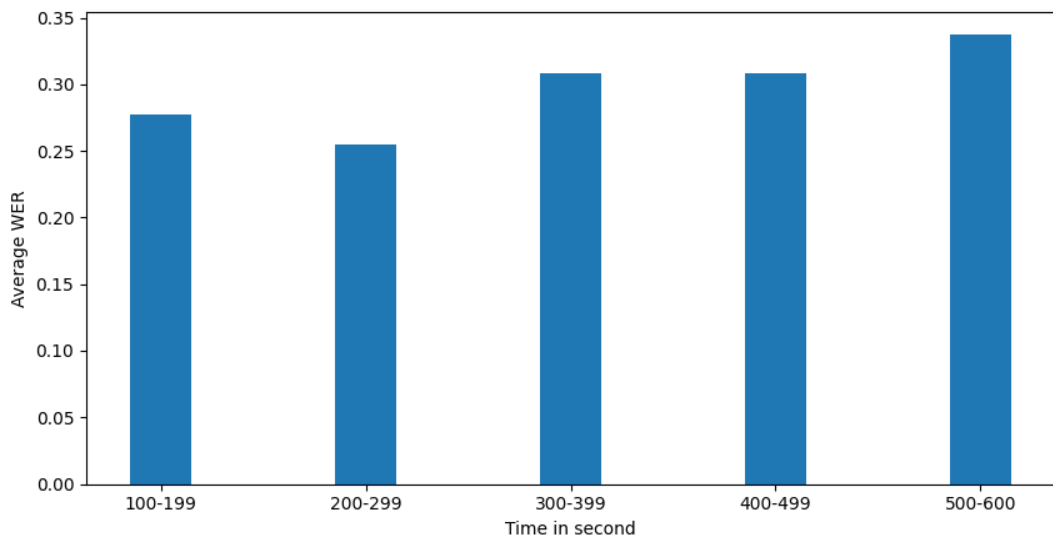


Figure 6.2: Bar plot of WER. We present the average WER in bar plot starting from 100 seconds up to 600 seconds. For each hundred seconds we calculate average WER and present it in the bar plot.

In figure 6.3 we present graphical view of average CER calculated for each hundred

seconds. From one hundred seconds up to six hundred seconds for each hundred seconds we calculated the average CER values. It shows the CER values depends on time of the data but not that much. Rather it more rely upon the clarity of character speech recognition.

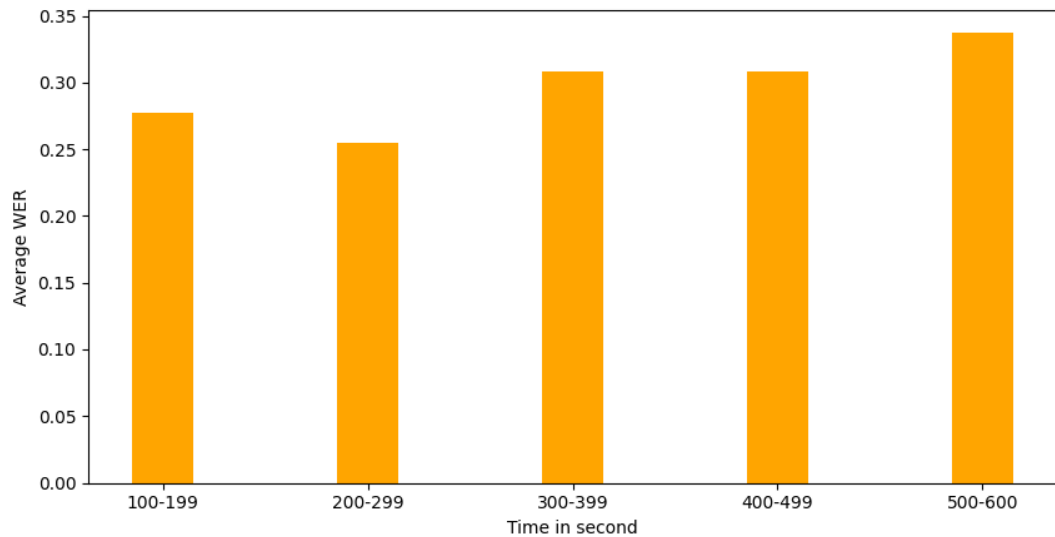


Figure 6.3: Bar plot of CER. We present the average CER in bar plot starting from 100 seconds up to 600 seconds. For each hundred seconds we calculate average CER and present it in the bar plot.

In figure 6.4 we present graphical view of average CER calculated for each hundred seconds. From one hundred seconds up to six hundred seconds for each hundred seconds we calculated the average CER values. It shows the CER values depends on time of the data but not that much. Rather it more rely upon the clarity of character speech recognition.

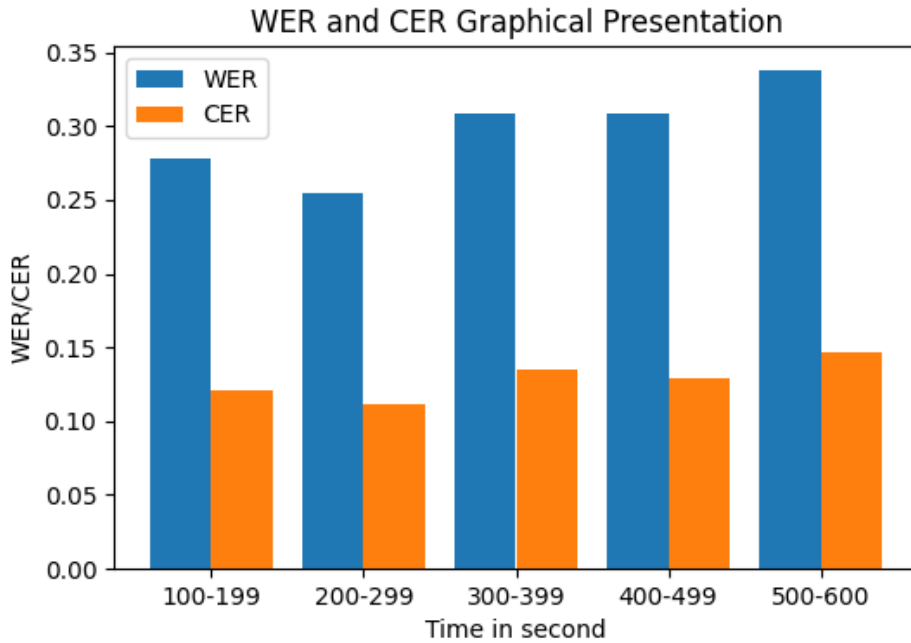


Figure 6.4: Bar plot showing WER and CER value side by side.

Table 6.4 contains the keywords that were extracted from the speech recognition transcript and represented graphically. Using our model, we put some spontaneous audiovisual content through it, and the keywords and phrases that we extracted are presented in the table below. There are single words, double words, and triple words in this list of keywords that are most frequent to the material itself. It is undeniable that the keywords are extremely diverse. This is due to the fact that, when extracting the keywords and phrases, we apply the Max Sum Similarity and the Maximal Marginal Relevance algorithms.

Table 6.4: Extracted keywords for five randomly chosen audiovisual file.

File Name	Extracted Keywords
Python: integers tutorial	Integers , Python, Numbers, Numeric, Decimal, Numeric data, Python integers, represented integers floats, numeric data python
Remove background from image using Photoshop	Photoshop, background, image, lasso tool, photoshop open image, photoshop background image
Wrapping a gift with paper wrapper	Paper, wrapping, package, wrapping packages, paper seam, wrapping packages small
Free Climber Emily Harrington Talks About Alex Honnold	Rogan, climbing, joe rogan, alex,climb, rogan experience, know climbing style,climbing placing gear
Grit: the power of passion and perseverance — Angela Lee Duckworth	Iq, intelligence, grades, teaching, math, iq scores, graders math, education better understanding, stratospheric iq scores, strongest intuitions need

There are single words, double words, and triple words in this list of keywords that are most frequent to the material itself. It is undeniable that the keywords are extremely diverse. This is because we apply the Max Sum Similarity and the Maximal Marginal Relevance algorithms. Though there exist variety but all of the keywords are relatively similar to the content because they are selected from a bag of words based on their cosine similarity to the text. As a result, these keywords are reliable for labeling content and grouping content against them in the database.

Table 6.5: Extracted keywords with relevance of randomly chosen audiovisual file.

File Name	Relevance of keywords
Python: integers tutorial	Integers (0.4818),Numbers (0.5416) Numeric (0.4846) Python integers (0.6303)
Remove background from image using Photoshop	Photoshop (0.3543), background (0.4238) lasso tool (0.4109) photoshop background image (0.4874)
Wrapping a gift with paper wrapper	Paper (0.4343), wrapping (0.4266) wrapping packages (0.4683) wrapping packages small (0.4981)
Free Climber Emily Harrington Talks About Alex Honnold	Rogan (0.3025), climb (0.4124) climbing (0.3188) rogan experience (0.3822)
Grit: the power of passion and perseverance — Angela Lee Duckworth	Iq (0.4874), intelligence (0.3968) graders math (0.4555) stratospheric iq scores (0.6353)

In table 6.5 we show four most relevant keywords for five randomly chosen audio-video file.

After the keywords are extracted, we label each audio-video files with the extracted keywords. Finally, we cluster the audio-video files based on the labelling keywords and for this task we have used K-means clustering. We calculated the sum of square

error values for each clustering and added all the sum of square error values together to get the final sum of square error value. After this, we used the elbow method to determine the k number, otherwise total clustering number. We take a small sample size of the dataset for clustering. After calculating the sum of square error and elbow method, the value of k we get is 53. That means our small sample size of the total audio-video dataset is cluster into 53 groups.

Chapter 7

Contribution, Challenges, Future work

In this section we have talked about the contributions of the paper, the challenges we have faced all through out our research and the future research possibilities.

7.1 Our contribution

In our research, we tried different paths and experiments in speech recognition and keywords extractions to construct an efficient and well performed model. The contributions of this paper is as follows:

1. We proposed a two stage model AVCL, that takes any audiovisual datas as input and labels it according to the content of the file and clusters them.
2. With our model AVCL, any unlabel or unnamed audio-video file can be labeled as to its content.
3. We provide an overview of speech recognition and keywords extraction using NLP in this paper.
4. We fine-tuned the wav2vec2 model and achieved better performance than many other versions of wav2vec2.
5. We have provided a model that can perform speech recognition on low resources and label the content.
6. There are many audiovisual dataset scatters all over. We have provided a collection of audiovisual datasets that we came across throughout our research so that other researchers might find it easy to collect these audiovisual datasets.

7.2 Challenges

Throughout our research, we encountered many difficulties. Some of these challenges limited our research and some we overcame in our research. The major challenges

we faced in our research are given below:

1. The major setback that we faced in our research is shortage of GPU's when we tried to train our model for speech recognition. Google uses Tesla K80 GPU for performing speech recognition but we cannot afford that due excessive cost. So we tried training the model in low GPUs but that did not provide any good results so we ended up using a pre-trained model.
2. We faced some difficulties while trying to evaluate longer audiovisual files for the shortage of graphics memory. So we divided the longer files into many chunks.
3. While collecting datasets, we faced difficulties as many adequate quality datasets are not available on open sources and there are no proper guidelines to acquire these licensed datasets. Even when we approached many organizations for their datasets, we did not receive any responses from them.
4. While speech recognition, the transcript we got did not have any punctuation and as a result we could not differentiate the sentences of the conversations.
5. When speech inputs deviate significantly from the norm, it can be difficult to work with. Accents, in particular, can be a significant difficulty. While systems are improving, there is still a significant disparity in their capacity to interpret different dialects of English, such as American or Indian English.
6. We started and completed our whole research during an unprecedented time of COVID pandemic. Due to this we sometimes could not work as a team and there are many resources that we could not get. So that is challenge we faced as team in this study.

7.3 Future work

While doing our study, we foreseen many new possible research areas closely related to our topic as well as with future research on our study a better performance can be achieved. Our experiment can serve as benchmarks for these future researches. Some of the future research that can be done are as follows:

1. Our model works on labeling saved conversations and further work on this can make a real time model which will transcript the conversation on the go and label the conversation as soon as it will end.
2. Further experiments with high end resources on our model and topic can achieve superior results.
3. In our work we only focused on labeling the content as to its topic but we did not focus on the emotions of the conversation. For future study, these two topics can be merged together for labeling the content as to its topic and the emotions of the speakers.

4. We performed our study based on the English language, so research on other languages can be done also.
5. For speech recognition we used wav2vec2 and for keyword extraction we used BERT model and achieved a fairly good result but future research can be done by using other models for speech recognition and keyword extraction which can give better results.

Chapter 8

Conclusion

In today's world of information and communication, audiovisual data is the most widely utilized and distributed on the internet. Because much of the audiovisual data is not labeled correctly, it is difficult to find the appropriate audiovisual data. So in this paper we proposed a two-stage model in this paper that will accept any audiovisual conversations as input and, using the concepts of deep neural networks, machine learning, and natural language processing to recognize the speeches of the conversations, transform it into a text document, and extract the keywords to label each conversation in accordance with its contextual setting. Our model correctly recognizes and labels any audiovisual speeches, allowing us to group them together for further analysis. The pre-trained models we used and fine-tuned surpass many other speech recognition models in terms of accuracy, and our model is capable of extracting keywords that are most relevant to the content and dialogue, which is enhanced by BERT's power, which yields excellent results. Because we concentrated solely on the content of the conversion and only on the English language, we opened the door to a plethora of other potential future researches. Further research into our topic has the potential to propel us to greater heights in the fields of speech recognition and keyword extraction using NLP. The outcomes of our model and experiment can serve as a baseline for all future work that is yet to be done. As a result of this study, we hope that other researchers will be inspired to work in the same field as ours in order to progress the fields of speech recognition and natural language processing, and that our model will serve as a stepping stone for future undertakings for the benefit of humanity.

References

- [1] K. R. Murphy and W. H. House, “The importance and preservation of the thoreau society’s audio and video media,” 2008.
- [2] L. BARSEGHYAN, “The role and importance of audio-visual aids in teaching,” 2015.
- [3] D. Hu, F. Nie, and X. Li, “Deep multimodal clustering for unsupervised audiovisual learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9248–9257.
- [4] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, and D. Tran, “Self-supervised learning by cross-modal audio-video clustering,” vol. 2020-December, 2020.
- [5] C. Comito, A. Forestiero, and C. Pizzuti, “Word embedding based clustering to detect topics in social media,” in *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, IEEE, 2019, pp. 192–199.
- [6] J. Hu, S. Li, Y. Yao, L. Yu, G. Yang, and J. Hu, “Patent keyword extraction algorithm based on distributed representation for patent classification,” *Entropy*, vol. 20, no. 2, p. 104, 2018.
- [7] I. Gagliardi and M. T. Artese, “Semantic unsupervised automatic keyphrases extraction by integrating word embedding with clustering methods,” *Multimodal Technologies and Interaction*, vol. 4, 2 2020, ISSN: 24144088. DOI: 10.3390/mti4020030.
- [8] E. Chu and D. Roy, “Audio-visual sentiment analysis for learning emotional arcs in movies,” vol. 2017-November, 2017. DOI: 10.1109/ICDM.2017.100.
- [9] A. Preciado-Grijalva and R. F. Brena, “Speaker fluency level classification using machine learning techniques,” *arXiv preprint arXiv:1808.10556*, 2018.
- [10] A. Ephrat, I. Mosseri, O. Lang, *et al.*, “Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation,” *ACM Transactions on Graphics*, vol. 37, 4 2018, ISSN: 15577368. DOI: 10.1145/3197517.3201357.
- [11] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno, and T. Ogata, “Audio-visual speech recognition using deep learning,” *Applied Intelligence*, vol. 42, 4 2015, ISSN: 15737497. DOI: 10.1007/s10489-014-0629-7.
- [12] M. Almousa, R. Benlamri, and R. Khoury, “Nlp-enriched automatic video segmentation,” vol. 2018-May, 2018. DOI: 10.1109/ICMCS.2018.8525880.

- [13] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, “Deep audio-visual speech recognition,” *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [14] T. Makino, H. Liao, Y. Assael, *et al.*, “Recurrent neural network transducer for audio-visual speech recognition,” in *2019 IEEE automatic speech recognition and understanding workshop (ASRU)*, IEEE, 2019, pp. 905–912.
- [15] H. Liao, E. McDermott, and A. Senior, “Large scale deep neural network acoustic modeling with semi-supervised training data for youtube video transcription,” in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, IEEE, 2013, pp. 368–373.
- [16] B. Shillingford, Y. Assael, M. W. Hoffman, *et al.*, “Large-scale visual speech recognition,” *arXiv preprint arXiv:1807.05162*, 2018.
- [17] T. Denton, A. Luebs, M. Chinen, *et al.*, “Handling background noise in neural speech generation,” in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, IEEE, 2020, pp. 667–671.
- [18] C. Yi, S. Zhou, and B. Xu, “Efficiently fusing pretrained acoustic and linguistic encoders for low-resource speech recognition,” *IEEE Signal Processing Letters*, vol. 28, pp. 788–792, 2021.
- [19] S. Khurana, A. Laurent, and J. Glass, “Magic dust for cross-lingual adaptation of monolingual wav2vec-2.0,” *arXiv preprint arXiv:2110.03560*, 2021.
- [20] S. Kessler, B. Thomas, and S. Karout, “Continual-wav2vec2: An application of continual learning for self-supervised automatic speech recognition,” *arXiv preprint arXiv:2107.13530*, 2021.
- [21] C. Yi, J. Wang, N. Cheng, S. Zhou, and B. Xu, “Applying wav2vec2. 0 to speech recognition in various low-resource languages,” *arXiv preprint arXiv:2012.12121*, 2020.
- [22] C. Abi Chahine, N. Chaignaud, J.-P. Kotowicz, and J.-P. Pécuchet, “Context and keyword extraction in plain text using a graph representation,” in *2008 IEEE International Conference on Signal Image Technology and Internet Based Systems*, IEEE, 2008, pp. 692–696.
- [23] E. Riloff *et al.*, “Automatically constructing a dictionary for information extraction tasks,” in *AAAI*, Citeseer, vol. 1, 1993, pp. 2–1.
- [24] F. Ciravegna, “2, an adaptive algorithm for information extraction from web-related texts,” in *In Proceedings of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*, Citeseer, 2001.
- [25] T. R. Leek, “Information extraction using hidden markov models,” M.S. thesis, Citeseer, 1997.
- [26] J. Lafferty, A. McCallum, and F. C. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” 2001.
- [27] P. J. Hayes and S. P. Weinstein, “Construe/tis: A system for content-based indexing of a database of news stories,” in *IAAI*, vol. 90, 1990, pp. 49–64.
- [28] Z. Zheng, “Naive bayesian classifier committees,” in *European Conference on Machine Learning*, Springer, 1998, pp. 196–207.

- [29] Y. Li, K. Bontcheva, and H. Cunningham, “Svm based learning system for information extraction,” in *International Workshop on Deterministic and Statistical Methods in Machine Learning*, Springer, 2004, pp. 319–339.
- [30] A. Vinnarasu and D. V. Jose, “Speech to text conversion and summarization for effective understanding and documentation,” *International Journal of Electrical and Computer Engineering*, vol. 9, no. 5, p. 3642, 2019.
- [31] H. Shukla and M. Kakkar, “Keyword extraction from educational video transcripts using nlp techniques,” in *2016 6th International Conference-Cloud System and Big Data Engineering (Confluence)*, IEEE, 2016, pp. 105–108.
- [32] D. Sahrawat, D. Mahata, H. Zhang, *et al.*, “Keyphrase extraction as sequence labeling using contextualized embeddings,” *Advances in Information Retrieval*, vol. 12036, p. 328, 2020.
- [33] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, “Attention is all you need,” in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
- [34] M. Tang, P. Gandhi, M. A. Kabir, C. Zou, J. Blakey, and X. Luo, “Progress notes classification and keyword extraction using attention-based deep learning models with bert,” *arXiv preprint arXiv:1910.05786*, 2019.
- [35] Y. Lim, D. Seo, and Y. Jung, “Fine-tuning bert models for keyphrase extraction in scientific articles,” *JOURNAL OF ADVANCED INFORMATION TECHNOLOGY AND CONVERGENCE*, vol. 10, no. 1, pp. 45–56, 2020.
- [36] H. Malm, M. Oskarsson, E. Warrant, P. Clarberg, J. Hasselgren, and C. Lejdfors, “Adaptive enhancement and noise reduction in very low light-level video,” in *2007 IEEE 11th International Conference on Computer Vision*, IEEE, 2007, pp. 1–8.
- [37] G.-Z. Yang, P. Burger, D. N. Firmin, and S. Underwood, “Structure adaptive anisotropic image filtering,” *Image and Vision Computing*, vol. 14, no. 2, pp. 135–145, 1996.
- [38] G. Morrone, S. Bergamaschi, L. Pasa, L. Fadiga, V. Tikhonoff, and L. Badino, “Face landmark-based speaker-independent audio-visual speech enhancement in multi-talker environments,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 6900–6904.
- [39] S.-Y. Chuang, H.-M. Wang, and Y. Tsao, “Improved lite audio-visual speech enhancement,” *arXiv preprint arXiv:2008.13222*, 2020.
- [40] R. Giri, U. Isik, and A. Krishnaswamy, “Attention wave-u-net for speech enhancement,” in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, IEEE, 2019, pp. 249–253.
- [41] R. Lu, Z. Duan, and C. Zhang, “Audio-visual deep clustering for speech separation,” *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 27, 11 2019, ISSN: 23299304. DOI: 10.1109/TASLP.2019.2928140.
- [42] J. MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Oakland, CA, USA, vol. 1, 1967, pp. 281–297.

- [43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [44] A. Baevski, S. Schneider, and M. Auli, “Vq-wav2vec: Self-supervised learning of discrete speech representations,” *arXiv preprint arXiv:1910.05453*, 2019.
- [45] A. Baevski, M. Auli, and A. Mohamed, “Effectiveness of self-supervised pre-training for speech recognition,” *arXiv preprint arXiv:1911.03912*, 2019.
- [46] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.
- [47] D. Hendrycks and K. Gimpel, “Gaussian error linear units (gelus),” *arXiv preprint arXiv:1606.08415*, 2016.
- [48] Y. Liu, M. Ott, N. Goyal, *et al.*, “Roberta: A robustly optimized bert pre-training approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [49] H. Jegou, M. Douze, and C. Schmid, “Product quantization for nearest neighbor search,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2010.
- [50] M. Grootendorst, *Keybert: Minimal keyword extraction with bert*. Version v0.3.0, 2020. DOI: 10.5281/zenodo.4461265. [Online]. Available: <https://doi.org/10.5281/zenodo.4461265>.
- [51] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, “Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers,” *arXiv preprint arXiv:2002.10957*, 2020.
- [52] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2015, pp. 5206–5210.
- [53] M. Cooke, J. Barker, S. Cunningham, and X. Shao, “An audio-visual corpus for speech perception and automatic speech recognition,” *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006.
- [54] W. Kay, J. Carreira, K. Simonyan, *et al.*, “The kinetics human action video dataset,” *arXiv preprint arXiv:1705.06950*, 2017.
- [55] J. F. Gemmeke, D. P. Ellis, D. Freedman, *et al.*, “Audio set: An ontology and human-labeled dataset for audio events,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2017, pp. 776–780.
- [56] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: A large-scale speaker identification dataset,” *arXiv preprint arXiv:1706.08612*, 2017.
- [57] K. J. Piczak, “Esc: Dataset for environmental sound classification,” in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1015–1018.
- [58] Y. Aytar, C. Vondrick, and A. Torralba, “Soundnet: Learning sound representations from unlabeled video,” *Advances in neural information processing systems*, vol. 29, pp. 892–900, 2016.

- [59] G. Wichern, J. Antognini, M. Flynn, *et al.*, “Wham!: Extending speech separation to noisy environments,” *arXiv preprint arXiv:1907.01160*, 2019.
- [60] N. Harte and E. Gillen, “Tcd-timit: An audio-visual corpus of continuous speech,” *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 603–615, 2015.
- [61] H. Kuwabara, K. Takeda, Y. Sagisaka, S. Katagiri, S. Morikawa, and T. Watanabe, “Construction of a large-scale japanese speech database and its management system,” in *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, 1989, pp. 560–563.
- [62] T. Afouras, J. S. Chung, and A. Zisserman, “Lrs3-ted: A large-scale dataset for visual speech recognition,” *arXiv preprint arXiv:1809.00496*, 2018.
- [63] M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler, “Movieqa: Understanding stories in movies through question-answering,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4631–4640.
- [64] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, “The million song dataset,” 2011.
- [65] T. Chen, D. Borth, T. Darrell, and S.-F. Chang, “Deepsentibank: Visual sentiment concept classification with deep convolutional neural networks,” *arXiv preprint arXiv:1410.8586*, 2014.
- [66] S. Zhou, Y. Zhao, S. Xu, B. Xu, *et al.*, “Multilingual recurrent neural networks with residual learning for low-resource speech recognition,” in *INTER-SPEECH*, 2017, pp. 704–708.
- [67] S. Zhou, S. Xu, and B. Xu, “Multilingual end-to-end speech recognition with a single transformer on low-resource languages,” *arXiv preprint arXiv:1806.05059*, 2018.
- [68] Q. Xu, T. Likhomanenko, J. Kahn, A. Hannun, G. Synnaeve, and R. Collobert, “Iterative pseudo-labeling for speech recognition,” *arXiv preprint arXiv:2005.09267*, 2020.