

# Construct a Customer Database from PDF Bank statements using Python Programming and Microsoft SQL

by

Bikash Kumar Nandi  
17366002

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

Department of Computer Science and Engineering  
Brac University  
June 2021

© 2021. Brac University  
All rights reserved.

# Declaration

It is hereby declared that

1. The thesis submitted is my 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:**



---

Bikash Kumar Nandi  
17366002

# Approval

The thesis/project titled “Construct a Customer Database from PDF Bank statements using Python Programming and Microsoft SQL” submitted by

1. Bikash Kumar Nandi (17366002)

Of Spring, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of M.Engg. in Computer Science and Engineering on June 6, 2021.

## Examining Committee:

Supervisor:  
(Member)



---

Dr. Jia Uddin  
Associate Professor (On leave)  
Department of Computer Science and Engineering  
BRAC University  
Assistant Professor (Research Track)  
Technology Studies Department  
Woosong University, Daejeon, South Korea.

Program Coordinator:  
(Member)



---

Dr. Amitabha Chakrabarty  
Associate Professor, Post Graduate Coordinator  
Department of Computer Science and Engineering  
BRAC University

Departmental Head:  
(Chair)



---

Sadia Hamid Kazi, Ph.D.  
Chairperson  
Department of Computer Science and Engineering  
BRAC University

## **Ethics Statement**

The project submitted is my original work, which has not been previously published elsewhere. Along with that, the paper is not currently being considered for publication elsewhere

## Abstract

This report proposes a model of extracting customers' transactions information from pdf Bank Account Statement and stores result-set into a customer Microsoft SQL (MsSQL) database for further automated analysis. In financial sector, it is very important to analysis bank account statement properly to measure the creditworthiness for credit approval. To achieve this target, a credit analyst needs to spend a significant time for manual analysis which leads to delay credit approval and sometimes inaccurate analysis diverts to take wrong approval. So, at present, automated bank account statement analysis is a big demand in the financial sector. This model will overcome the aforementioned limitations and serve the current market demand. For targeting to achieve this desired goal, the whole process has been divided into 4 basic segments. The first segment entails converting pdf to text by using a python library (pdftotext), the second one emphasis on correction raw text file (.txt) data by removing unnecessary characters and spaces and do formatting as per need, the third segment consists of parsing formatted text (.txt) and retrieving desired transactional information, and finally the fourth segment stores the desired information into a customer database.

**Keywords:** Bank Account Statement; Customer Database; Microsoft SQL; Financial Sector; Credit Approval; Python; pdftotext

# Dedication

I dedicate this project to my respected teachers for mentoring and encouraging me always and also to my parents and wife for inspiring me.

## **Acknowledgement**

My heartiest gratitude to the Almighty, my parents and my wife for empowering me to complete my Postgraduate studies. I am grateful to my supervisor, Dr. Jia Uddin for his sincere guidance and tireless help and his encouragement. I would like to thank post graduate coordinator, Dr. Amitabha Chakraborty for granting and guidance me to complete my Postgraduate studies. I am also wish to thank CSE department of BRAC University for giving opportunity and support me to accomplish my degree.

# Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iii
Abstract	iv
Dedication	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	ix
List of Tables	x
Nomenclature	xi
<b>1 Introduction</b>	<b>1</b>
1.1 Background Analysis . . . . .	1
1.2 Motivation . . . . .	1
1.3 Objective . . . . .	2
1.4 Contributions . . . . .	2
1.5 Project Outline . . . . .	2
<b>2 Related Work</b>	<b>3</b>
<b>3 Proposed Model</b>	<b>4</b>
3.1 Brief Overview . . . . .	4
3.2 Converting PDF to Text . . . . .	4
3.3 Correction Raw Text . . . . .	6
3.4 Parsing Formatted Text . . . . .	7
3.5 Preserving Statement Info into Database . . . . .	8
3.6 Algorithm . . . . .	9
<b>4 Experimental setup, Result analysis and Comparative analysis</b>	<b>15</b>
4.1 Data Collection . . . . .	15
4.2 Result Analysis . . . . .	16
4.3 Comparative Analysis . . . . .	17



<b>5 Conclusion and Future Work</b>	<b>18</b>
5.1 Conclusion . . . . .	18
5.2 Future Work . . . . .	18
<b>Bibliography</b>	<b>20</b>

# List of Figures

3.1	Block Diagram of the Proposed Model . . . . .	4
3.2	Sample Bank Account Statement (pdf version) . . . . .	5
3.3	Converted Raw Text (.txt) of Sample Bank Account Statement . . . . .	5
3.4	Formatted Text File (.txt) of Sample Bank Account Statement . . . . .	6
3.5	Statement Summary and Details of Sample Bank Account Statement . . . . .	7
3.6	Statement Summary from Database Table . . . . .	8
3.7	Statement Transactions Details from Database Table . . . . .	8
4.1	List of account statements . . . . .	15
4.2	Statement Summary info in database table . . . . .	16
4.3	Sample pdf statement by mentioning problem area . . . . .	16
4.4	Details Info of sample pdf statement in database table . . . . .	17

# List of Tables

# Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

*MsSQL* Microsoft Structured Query Language

*NPL* Non-Performing Loan

*PDF* Portable Document Format

# Chapter 1

## Introduction

### 1.1 Background Analysis

A bank statement is a financial transactions summary document for the specific period of any account which is provided by the financial institution. Currently, most of the financial institutions provided bank statement to customers as pdf version[1] through e-mail or made it available to download statement through internet banking as per customers' need.

To avail any amount of credit from any financial institution, customer must need to submit existing bank accounts' statements to the financial institution.

Now a days, the financial sector is looking to approve credit proposal swiftly and accurately by targeting to earn more profit and reduce non-performing loan (NPL). Besides, this sector is getting so competitive due to numerous companies and their offered products' are very close. Therefore, companies are eagerly looking to customer satisfaction and customer retention by providing better service. Consequently, currently most of the financial companies are focusing to do digitalization[5], [11] their services through automation.

In this regard, automated bank account statement extraction can have a significant role to take credit approval decision swiftly and accurately for the growth and achievement of companies as well as better service to customers. To achieve this target, through automation pdf bank statement needs to be extracted into text file [24] and preserved into a customer database for detail analysis.

### 1.2 Motivation

Manually bank account statement analysis by credit analyst is very time consuming which leads to delay credit approval and sometimes there is a chance to do mistake which leads to non-performing loan, by which companies experience huge loss [21].

So, if bank account statement's information can be extracted properly and store transactions information[26] into a customer database, by using reporting tool necessary analytical reports and graphs can be presented to Credit analyst. Consequently, the whole analysis will be faster than manual process as well as it can be

avoid to manual mistake.

## 1.3 Objective

The desired goal of this project is to extract pdf[8], [15], [26] bank account statement into text file by using python library (pdftotext)[25] which is further formatted and processed. As a result, statement summary and transactions details information are extracted which is finally stored into a customer database.

## 1.4 Contributions

There are few challenges after converting pdf statement to text because the converted text doesn't follow any format[16] so that it is quite difficult to extract required information properly. So we focus to apply logics on converted text[5] to do format in such a way so that unnecessary texts can be easily identified and bypassed and grabbed only desired information[20].

Moreover, in the formatted text, we found statement summary info available in the first page and last page also, so we developed a logic to catch summary info in such a way which works properly for any size of statement[6]. In addition, in transaction details section, we noticed only first row shows the forward balance where there is no transaction date and sometimes transaction's particular splits into new row and transaction header info is also available in all pages. Consequently, we built few logics by targeting to grab transactions details properly[4].

## 1.5 Project Outline

The project report has been structured in the following way:

Chapter 2 comprises the related work

Chapter 3 proposed model

Chapter 4 contains Experimental setup, Result analysis and Comparative analysis

Chapter 5 contains conclusion and future work

# Chapter 2

## Related Work

PDF format is very useful because it stores the structure of the document across any platforms [14], [15]. Though, PDF allows the shield of the look of any document, it does may not be possible to do logical representation of the text accurately [3], [24]. After extraction text from PDF files, firstly it is noticed that text streams can resemble to many objects: a character, a word, a partial word, a line etc. Secondly sometimes extracted text order does not follow the order of reading text [20]. So, during text extraction from a PDF file, it is quite vital to recognize reconstruction component of a word and order component of a reading sequence. To analysis PDF data, at first, PDF[1] needs to be converted into HTML and after that, it is very important to do detail analysis by using the HTML tags[7]. After conversion into HTML format, it needs to focus on all the information which is related to text formatting because it is important for detail analysis. There are a variety of conversion tools to convert from PDF to HTML document. The quality of the tools have been evaluated by checking the reading text and after conversion, structural loss related with each tool[19]. There are few additional activities in PDF to HTML text detection approaches that retain layout and font information, table detection, extraction and annotation[18] and analysis using white spaces [22]. To analyze PDF layout and content properly, HTML conversion technique is a very good approach[23].

# Chapter 3

## Proposed Model

### 3.1 Brief Overview

In this proposed model, the whole system has been designed in very efficient approach and segmented into 4 parts. The first segment consists of converting bank account statement from pdf to text by using a popular python[12] library (pdftotext) [25]. After conversion, the second segment mainly focuses to remove unwanted character, space, line, etc.[6] and formats in such a way so that necessary information can be extracted very easily and quickly [4]. Later, the third segment gives emphasis to apply logics in formatted text for retrieving statement summary and transaction details. Finally, the fourth segment is mainly responsible to establish database connectivity for storing statement information into database. Figure 3.1 depicts the whole process in block diagram.

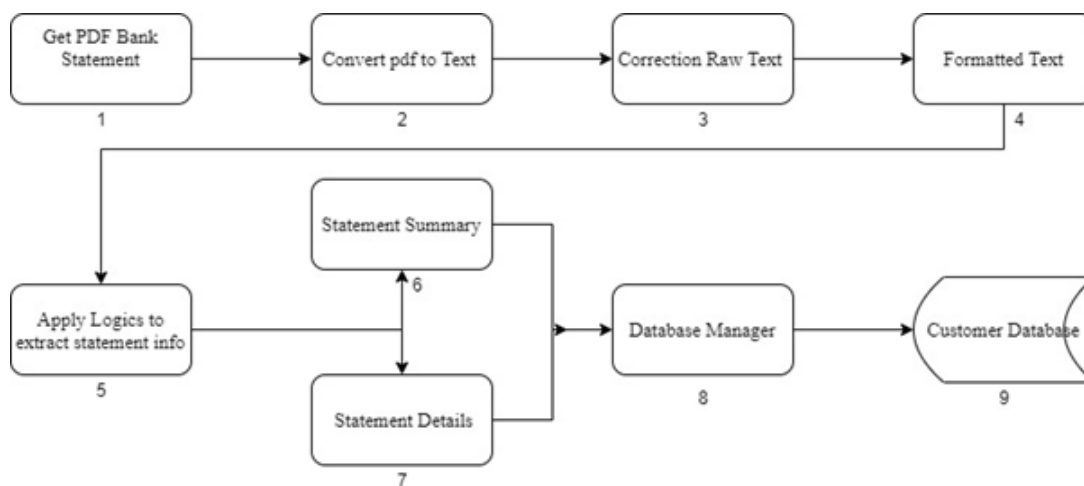


Figure 3.1: Block Diagram of the Proposed Model

### 3.2 Converting PDF to Text

For converting pdf bank account statement to text, a popular python[12] library (pdftotext) has been used which can convert password protected pdf file easily and quickly[25]. Interestingly, we just need to call pdftotext class by passing parameters such as file password, input file path and output file path. We observe that 10 to 15



pages statement can be converted within a few seconds and stored raw text file in output directory. Figure 3.2 shows Sample Bank Account Statement (pdf version) and Figure 3.3 describes Converted Raw Text of Sample Bank Account Statement.

Issue Date : October 01, 2020

**STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 30-Jun-2020**

DATE	PARTICULARS	CHEQ. NO	WITHDRAW	DEPOSIT	BALANCE
	Balance Forward		0.00	0.00	18,235.99
01-Jul-2019	HF YRLY AC MNT FEE-2019		100.00	0.00	18,135.99
01-Jul-2019	VAT ON HF YRLYAC MNT FEE-2019		15.00	0.00	18,120.99
02-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002		3,500.00	0.00	14,620.99
07-Jul-2019	CWOR/HAZICAMP DHAKA BD /15011016190002		10,000.00	0.00	4,620.99
07-Jul-2019	CWOR/HAZICAMP DHAKA BD /15011016190002		2,500.00	0.00	2,120.99
07-Jul-2019	IB/EFT/018662/OC/NBL/1067001657509		2,000.00	0.00	120.99
07-Jul-2019	IB/024176/self		0.00	30,000.00	30,120.99
07-Jul-2019	CWOR/DUTCH-BANGLA BANK LTD. DHAKA/15011016190002		16,515.00	0.00	13,605.99
08-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002		4,800.00	0.00	8,805.99
11-Jul-2019	INH/006774/BKS/OT/01734479980		3,001.00	0.00	5,804.99
11-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002		5,500.00	0.00	304.99
14-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002		500.00	0.00	104.99
15-Jul-2019	IB/028289/Personal		0.00	15,000.00	15,104.99
15-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002		4,500.00	0.00	10,604.99
17-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002		2,500.00	0.00	8,104.99
18-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002		2,500.00	0.00	5,604.99
21-Jul-2019	CWOR/SALAHAMPUR DHAKA BD /15011016190002		4,515.00	0.00	1,089.99

Page 1 of 10

Figure 3.2: Sample Bank Account Statement (pdf version)

File Edit Format View Help

Account Type : SAVINGS ACCOUNT  
Currency Issue Date : 01 October 2020

STATEMENT OF	ACCOUNT	FOR THE PERIOD	OF	01-Jul-2019	30-Jun-2020	WITHDRAW	DEPOSIT	BALANCE
DATE	PARTICULARS		CHEQ. NO					
01-Jul-2019	Balance Forward					0.00	0.00	18,235.99
01-Jul-2019	HF YRLY AC MNT FEE-2019					100.00	0.00	18,135.99
01-Jul-2019	VAT ON HF YRLYAC MNT FEE-2019					15.00	0.00	18,120.99
02-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002					3,500.00	0.00	14,620.99
07-Jul-2019	CWOR/HAZICAMP DHAKA BD /15011016190002					10,000.00	0.00	4,620.99
07-Jul-2019	CWOR/HAZICAMP DHAKA BD /15011016190002					2,500.00	0.00	2,120.99
07-Jul-2019	IB/EFT/018662/OC/NBL/1067001657509					2,000.00	0.00	120.99
07-Jul-2019	IB/024176/self					0.00	30,000.00	30,120.99
07-Jul-2019	CWOR/DUTCH-BANGLA BANK LTD. DHAKA/15011016190002					16,515.00	0.00	13,605.99
08-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002					4,500.00	0.00	9,105.99
11-Jul-2019	INH/006774/BKS/OT/01734479980					3,001.00	0.00	6,104.99
11-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002					5,500.00	0.00	504.99
14-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002					500.00	0.00	104.99
15-Jul-2019	IB/028289/Personal					0.00	15,000.00	15,104.99
15-Jul-2019	CWOR/Anik Tower ATM DHAKA BD /15011016190002					4,500.00	0.00	10,604.99
17-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002					2,500.00	0.00	8,104.99
18-Jul-2019	CWOR/Anik Tower-2nd DHAKA BD /15011016190002					2,500.00	0.00	5,604.99
21-Jul-2019	CWOR/SALAHAMPUR DHAKA BD /15011016190002					4,515.00	0.00	1,089.99

Page 1 of 10

Figure 3.3: Converted Raw Text (.txt) of Sample Bank Account Statement

### 3.3 Correction Raw Text

After converting pdf file to text, many unnecessary characters and spaces are found in Raw text (.txt) file [20]. Therefore, it is quite difficult to apply any algorithm to retrieve necessary information[17]. That's why, few actions have been taken to remove unwanted characters and spaces and formatted text[13]. Consequently, formatted text file (.txt) has been produced. Figure 3.4 shows the Formatted Text of Sample Bank Account Statement.

```
File Edit Format View Help
| Account| Type| : SAVINGS| ACCOUNT
| Currency| : BDT
| Issue| Date| : October| 01, 2020
| STATEMENT| OF ACCOUNT| FOR THE| PERIOD| OF 01-Jul-2019| TO 30-Jun-2020

| DATE| PARTICULARS| CHQ.NO| WITHDRAW| DEPOSIT| BALANCE
| Balance| Forward| 0.00| 0.00| 18,235.89
| 01-Jul-2019| HF YRLY AC MNT FEE-2019| 100.00| 0.00| 18,135.89
| 01-Jul-2019| VAT ON HF YRLYAC| MNT| 15.00| 0.00| 18,120.89
| FEE-2019
| 02-Jul-2019| CWDR/Anik| Tower| ATM DHAKA| BD| 3,500.00| 0.00| 14,620.89
| /1501101613...002
| 07-Jul-2019| CWDR/HAZICAMP| DHAKA| BD| 10,000.00| 0.00| 4,620.89
| /1501101613...002
| 07-Jul-2019| CWDR/HAZICAMP| DHAKA| BD| 2,500.00| 0.00| 2,120.89
| /1501101613...002
| 07-Jul-2019| IB/EFT/018662/OC/NBL/1067001| 2,000.00| 0.00| 120.89
| 657509
| 07-Jul-2019| IB/024176/self| 0.00| 30,000.00| 30,120.89
| 07-Jul-2019| CWDR/DUTCH-BANGLA| BANK LTD.| 16,515.00| 0.00| 13,605.89
| DHAK/1501101613...002
| 09-Jul-2019| CWDR/Anik| Tower| ATM DHAKA| BD| 4,500.00| 0.00| 9,105.89
| /1501101613...002
| 11-Jul-2019| INH/006774/BKS/OT/0173447998| 3,001.00| 0.00| 6,104.89
| 0
| 11-Jul-2019| CWDR/Anik| Tower| ATM DHAKA| BD| 5,500.00| 0.00| 604.89
| /1501101613...002
| 14-Jul-2019| CWDR/Anik| Tower-2nd| DHAKA| BD| 500.00| 0.00| 104.89
| /1501101613...002
| 15-Jul-2019| IB/028339/Personal| 0.00| 15,000.00| 15,104.89
| 15-Jul-2019| CWDR/Anik| Tower| ATM DHAKA| BD| 4,500.00| 0.00| 10,604.89
| /1501101613...002
| 17-Jul-2019| CWDR/Anik| Tower-2nd| DHAKA| BD| 2,500.00| 0.00| 8,104.89
| /1501101613...002
| 18-Jul-2019| CWDR/Anik| Tower-2nd| DHAKA| BD| 2,500.00| 0.00| 5,604.89
| /1501101613...002
<
```

Figure 3.4: Formatted Text File (.txt) of Sample Bank Account Statement

### 3.4 Parsing Formatted Text

This is the third segment of entire process which is mainly responsible to retrieve necessary transactional information by scanning the whole formatted text file (.txt) data[6]. Firstly, blank line and page footer line have been bypassed and picked up rest of the lines from formatted text file[7]. Then, starting point and end point of transaction details have been identified and scanned in this area. One of the challenging parts is “Forwarded Balance” which is the starting line of transaction details but it’s different with other lines, so this line has been addressed separately. Another challenging part is sometimes transaction’s particular split into new line[2]. So, during scanning each line, its need to check that whether transactions particular is split into the next line.

Moreover, statement summary part stands just above of the starting point of transaction details and just above of end point[1]. Figure 3.5 displays the output of the Statement Summary and Details of Sample Bank Account Statement which are recognized in this segment.

```
C:\Users\bikash28131\AppData\Local\Programs\Python\Python37\python.exe E:/Python/Work/StatementPDFtoDB.  
*****Statement Summary*****  
"statement_period" : STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 30-Jun-2020  
"issue_date" : October 01, 2020  
"cid" : 01613  
"acc_no" : 1501101613002  
"acc_type" : SAVINGS ACCOUNT  
"currency" : BDT  
"total_withdraw" : 1,611,268.79  
"total_deposit" : 1,654,790.85  
"total_balance" : 61,757.95  
  
*****Statement Details*****  
['01-Jul-2019', 'Balance Forward', '0.00', '0.00', '18,235.89']  
['01-Jul-2019', 'HF YRLY AC MNT FEE-2019', '100.00', '0.00', '18,135.89']  
['01-Jul-2019', 'VAT ON HF YRLYAC MNT FEE-2019', '15.00', '0.00', '18,120.89']  
['02-Jul-2019', 'CWDR/Anik Tower ATM DHAKA BD /1501101613002', '3,500.00', '0.00', '14,620.89']  
['07-Jul-2019', 'CWDR/HAZICAMP DHAKA BD /1501101613002', '10,000.00', '0.00', '4,620.89']  
['07-Jul-2019', 'CWDR/HAZICAMP DHAKA BD /1501101613002', '2,500.00', '0.00', '2,120.89']  
['07-Jul-2019', 'IB/EFT/018662/OC/NBL/1067001 657509', '2,000.00', '0.00', '120.89']  
['07-Jul-2019', 'IB/024176/self', '0.00', '30,000.00', '30,120.89']  
['07-Jul-2019', 'CWDR/DUTCH-BANGLA BANK LTD. DHAK/1501101613002', '16,515.00', '0.00', '13,605.89']  
['09-Jul-2019', 'CWDR/Anik Tower ATM DHAKA BD /1501101613002', '4,500.00', '0.00', '9,105.89']  
['11-Jul-2019', 'INH/006774/BKS/OT/0173447998 0', '3,001.00', '0.00', '6,104.89']
```

Figure 3.5: Statement Summary and Details of Sample Bank Account Statement

### 3.5 Preserving Statement Info into Database

This segment focuses on to preserve transactions summary and details into database. At first, it establishes a connectivity with database. Then it executes SQL stored procedure by passing summary data. After successful insertion, a reference id is generated which is used to store details transactions. Later, similarly details transactions are inserted into database with reference id. Figure 3.6 and Figure 3.7 portray few Statement Summary and corresponding Transactions Details from Database Table.

ID	statement_period	issue_date	currency	acc_type	Account_No	CID	total_withdraw	total_deposit	total_balance
1	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 30-Jun-2020	2020-10-01	BDT	SAVINGS ACCOUNT	1501101613***002	01613***	1611268.79	1654790.85	61757.95
2	STATEMENT OF ACCOUNT FOR THE PERIOD OF 08-Nov-2018 TO 07-May-2019	2019-05-07	BDT	SAVINGS ACCOUNT	1501101613***001	01613***	4230646.46	4130200.62	157013.64
3	STATEMENT OF ACCOUNT FOR THE PERIOD OF 11-Feb-2020 TO 09-Aug-2020	2020-08-09	BDT	SAVINGS ACCOUNT	1501101613***002	01613***	910042.29	919150.00	17685.95

Figure 3.6: Statement Summary from Database Table

ID	tran_date	tran_particulars	withdraw	deposit	balance
1	2019-07-01	Balance Forward	0.00	0.00	18235.89
2	2019-07-01	HF YRLY AC MNT FEE-2019	100.00	0.00	18135.89
3	2019-07-01	VAT ON HF YRLYAC MNT FEE-2019	15.00	0.00	18120.89
4	2019-07-02	CWDR/Anik Tower ATM DHAKA BD /1501101613***002	3500.00	0.00	14620.89
5	2019-07-07	CWDR/HAZICAMP DHAKA BD /1501101613***002	10000.00	0.00	4620.89
6	2019-07-07	CWDR/HAZICAMP DHAKA BD /1501101613***002	2500.00	0.00	2120.89
7	2019-07-07	IB/EFT/018662/OC/NBL/1067001 657509	2000.00	0.00	120.89
8	2019-07-07	IB/024176/self	0.00	30000.00	30120.89
9	2019-07-07	CWDR/DUTCH-BANGLA BANK LTD. DHAK/1501101613***002	16515.00	0.00	13605.89
10	2019-07-09	CWDR/Anik Tower ATM DHAKA BD /1501101613***002	4500.00	0.00	9105.89
11	2019-07-11	INH/006774/BKS/OT/0173447998 0	3001.00	0.00	6104.89
12	2019-07-11	CWDR/Anik Tower ATM DHAKA BD /1501101613***002	5500.00	0.00	604.89
13	2019-07-14	CWDR/Anik Tower-2nd DHAKA BD /1501101613***002	500.00	0.00	104.89
14	2019-07-15	IB/028339/Personal	0.00	15000.00	15104.89
15	2019-07-15	CWDR/Anik Tower ATM DHAKA BD /1501101613***002	4500.00	0.00	10604.89
16	2019-07-17	CWDR/Anik Tower-2nd DHAKA BD /1501101613***002	2500.00	0.00	8104.89
17	2019-07-18	CWDR/Anik Tower-2nd DHAKA BD /1501101613***002	2500.00	0.00	5604.89
18	2019-07-21	CWDR/SHAJAHANPUR DHAKA BD DHA/1501101613***002	4515.00	0.00	1089.89
19	2019-07-22	CWDR/NIKUNJA APON SOMOY	1000.00	0.00	89.89
20	2019-07-23	IB/034585/Personal	0.00	10000.00	10089.89
21	2019-07-24	CWDR/ANIK TOWER 3RD DH A/1501101613***002	3500.00	0.00	6589.89
22	2019-07-28	CWDR/UTTARA JASIM UDDIN BR. DHAK/1501101613***002	4500.00	0.00	2089.89
23	2019-07-28	CWDR/Anik Tower ATM DHAKA BD /1501101613***002	2000.00	0.00	89.89

Figure 3.7: Statement Transactions Details from Database Table

## 3.6 Algorithm

Here is the algorithm of the whole process:

```
//Convert pdf to text
Procedure ConvertPDFToText (InPath, OutPath, FilePassword)
Begin

    Execute pdftotext FilePassword, InPath, OutPath

End

//Formatting RAW TEXT

Procedure FormatRAWTEXT (OutPath)
Begin

    SET textfile = READ(OutPath)

    SET textfile = by replacing special character (page break) by empty character
    from textfile

    SET textfile = by replacing large space by pipe char from textfile from textfile

    //Still remaining multi space will

    SET textfile = by replacing multi-space by single space from textfile

    RE-WRITE (textfile, OutPath)

End

Procedure ParseText (OutPath, MasterInfo[], DetailsInfo[])
Begin

    SET statementStart = 0

    SET statementEnd = 0

    SET lines[] =null

    SET txtFile=READ (OutPath)

    FOR line in txtFile
```

```

IF length of line equal to 1 or "Page" in line THEN
    Pass
ELSE
    Append line in lines array
END IF
END FOR
FOR row in range lines.len()
    IF "PARTICULARS" in lines[row] and statementStart = 0 THEN
        SET statementStart = row
    ELSE IF "Reward" in lines[row] and statementEnd = 0 THEN
        SET statementEnd = row-1
        Break;
    END IF
END FOR
//Pick Statement Summary
SET statementperiod = the value of lines[statementStart-1];
SET issuedate = the value of lines[statementStart - 2];
SET currency = the value of lines[statementStart - 3];
SET acctype = the value of lines[statementStart - 4];
SET accno = the value of lines[statementStart - 5];
SET refandcid = the value of lines[statementStart - 6];
SET ref = the value of refandcid[0] after split by ':'
SET cid = the value of refandcid[1]
After splitting lines[statementEnd] by pipe char

```

```

SET totalwithdraw = the value of lines[statementEnd][-3]

SET totaldeposit = the value of lines[statementEnd][-2]

SET totalbalance = the value of lines[statementEnd][-1]

Append [statementperiod, issuedate, currency, acctype,accno, ref, cid,
totalwithdraw, totaldeposit,totalbalance] in MasterInfo[] array

//Pick Statement transaction details

FOR row in range (statementStart, statementEnd)

    //Consider only balance forwarded line

    IF "PARTICULARS" not in lines[row] and "Balance" in lines[row] THEN

        SET startdate = the value of statementperiod by splitting pipe char and
        getting 2nd value from last

        SET transparticular = the value of lines[row] by splitting pipe char and
        getting 1nd value from starting

        SET withdraw = the value of lines[row] by splitting pipe char and
        getting 2nd value from starting

        SET deposit = the value of lines[row] by splitting pipe char and
        getting 3rd value from starting

        SET balance = the value of lines[row] by splitting pipe char and
        getting 4th value from starting

        SET balanceforward [] = [startdate, transparticular, withdraw,
        deposit, balance] this data set

        Append balanceforward in DetailsInfo[] array

    END IF

// consider other transactional lines

```

```

IF "PARTICULARS" not in lines[row] and "Balance" not in lines[row] THEN

    //Consider this line is solid transactional line

    IF length of lines[row] by splitting pipe char greater than 4 is true THEN

        SET txndate = the value of lines[row] by splitting pipe char and
        getting 1st value from starting

        SET balance = the value of lines[row] by splitting pipe char and
        getting 1st value from last

        SET deposit = the value of lines[row] by splitting pipe char and
        getting 2nd value from last

        SET withdraw = the value of lines[row] by splitting pipe char and
        getting 3rd value from last

        SET withdrawstartingposition = the value of starting position of
        withdraw

        SET particulars = the value of lines[row] from

            txndate to withdrawstartingposition
        //Check whether Trans particular split into new line

        IF length of lines[row+1] by splitting pipe char less than 3 is true THEN

            SET particulars = the value of particulars and the value of

            lines[row + 1]

        END IF

        Append [startdate, transparticular, withdraw, deposit,

        balance] in DetailsInfo[]

    END IF

END FOR

```



End

Procedure DatabaseManager (MasterInfo[], DetailsInfo[])

Begin //Establish connection with Database

Conn=DBConnection()

//Store Master info into Database

Execute Database SP by passing MasterInfo [] and

picking reference ID after

inserting data into master table in database

FOR row in DetailsInfo[]

Execute Database SP by passing row values with MasterID

END FOR

Close Conn

End

//Starting Position of the program

Function Main ()

Begin

SET InPath to Input file path

SET OutPath to Output file path

SET FilePassword to File password

//Convert pdf to text and generated text file will be saved in OutPath

Call ConvertPDFtoText (In Path, OutPath, File Password)

// Format RAW Text from OutPath

Call FormatRAWTEXT (OutPath)

```
//Parse Formatted Text  
Call ParseText (OutPath, MasterInfo [], DetailsInfo [])  
  
//Store statement info into database  
  
Call DatabaseManager (MasterInfo[], DetailsInfo[])  
  
End
```

# Chapter 4

## Experimental setup, Result analysis and Comparative analysis

### 4.1 Data Collection

Since transactional data poses many issues, in this experiment mainly different periods of my savings account statements of BRAC Bank Ltd have been used. Since the format of the statements of all customers in BRAC Bank is same, lack of other customers' statement doesn't do any worse impact in this current experiment. Because the goal of this experiment is to extract transactional information from pdf statement and store into a customer database perfectly. For this research, 12 statements of 2 accounts with a variety of statement periods are used. Figure 4.1 shows the list of account statements which have been used.













Name	Date modified	Type	Size
 BRAC_BANK_STATEMENT_0001_01Jan19_05May19.pdf	5/5/2019 6:04 PM	Adobe Acrobat D...	182 KB
 BRAC_BANK_STATEMENT_0001_01Jul19_30Jun20.pdf	11/21/2020 3:17 PM	Adobe Acrobat D...	186 KB
 BRAC_BANK_STATEMENT_0001_04Apr19_01Oct19.pdf	10/1/2019 7:34 PM	Adobe Acrobat D...	181 KB
 BRAC_BANK_STATEMENT_0001_08Nov18_07May19.pdf	5/7/2019 5:31 PM	Adobe Acrobat D...	185 KB
 BRAC_BANK_STATEMENT_0002_01Jul18_30Jun19.pdf	5/10/2021 3:39 PM	Adobe Acrobat D...	184 KB
 BRAC_BANK_STATEMENT_0002_01Jul19_22Aug19.pdf	8/22/2019 11:45 AM	Adobe Acrobat D...	179 KB
 BRAC_BANK_STATEMENT_0002_01Jul19_30Jun20.pdf	5/10/2021 3:28 PM	Adobe Acrobat D...	166 KB
 BRAC_BANK_STATEMENT_0002_01Mar18_20Mar19.pdf	3/20/2019 9:55 AM	Adobe Acrobat D...	182 KB
 BRAC_BANK_STATEMENT_0002_10May20_09Aug20.pdf	8/9/2020 9:13 AM	Adobe Acrobat D...	179 KB
 BRAC_BANK_STATEMENT_0002_11Feb20_09Aug20.pdf	8/9/2020 9:12 AM	Adobe Acrobat D...	182 KB
 BRAC_BANK_STATEMENT_0002_15Sep19_23Mar20.pdf	3/13/2020 2:37 PM	Adobe Acrobat D...	187 KB
 BRAC_BANK_STATEMENT_0002_16Oct19_30Oct19.pdf	10/30/2019 1:27 PM	Adobe Acrobat D...	176 KB

Figure 4.1: List of account statements

## 4.2 Result Analysis

During statement processing, at first password protected pdf statement is converted to text format by using python library pdftotext [7], as explain in section 3.2. After that, the necessary formatting is applied and saved back to the text file, as explain in section 3.3. Then, formatted text is parsed and extracted statement information, as explain in section 3.4 and finally extracted statement information is stored into a customer database, as explain in section 3.5. By applying the aforementioned approach, all 12 statements have been extracted successfully without any crash and preserved into the database properly. Statement information has been split into two categories: one – statement summary which has been stored into statement summary info table in database, second – statement details which has been stored into statement details info table by keeping relation id with summary info in database level. Accordingly, in summary info table, we have found 12 records with all proper information which depicts in figure 4.2.

ID	Statement Period	Issue date	Currency	Account Type	Account No.	Customer ID	Total Withdraw	Total Deposit	Total Balance
1	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jan-2019 TO 05-May-2019	2019-05-05	BDT	SAVINGS ACCOUNT	1501101613	001 01613	3871705.71	3923383.85	166613.64
2	STATEMENT OF ACCOUNT FOR THE PERIOD OF 08-Nov-2018 TO 07-May-2019	2019-05-07	BDT	SAVINGS ACCOUNT	1501101613	001 01613	4230646.46	4130200.62	157013.64
3	STATEMENT OF ACCOUNT FOR THE PERIOD OF 04-Apr-2019 TO 01-Oct-2019	2019-10-01	BDT	SAVINGS ACCOUNT	1501101613	001 01613	4414129.82	4677943.67	353705.03
4	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 30-Jun-2020	2020-11-21	BDT	SAVINGS ACCOUNT	1501101613	001 01613	6239460.82	6137524.16	26814.45
5	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Mar-2018 TO 20-Mar-2019	2019-03-20	BDT	SAVINGS ACCOUNT	1501101613	002 01613	919983.23	877837.32	5853.49
6	STATEMENT OF ACCOUNT FOR THE PERIOD OF 23-Jul-2019 TO 22-Aug-2019	2019-08-22	BDT	SAVINGS ACCOUNT	1501101613	002 01613	82515.00	83500.00	1074.89
7	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 22-Aug-2019	2019-08-22	BDT	SAVINGS ACCOUNT	1501101613	002 01613	145661.00	128500.00	1074.89
8	STATEMENT OF ACCOUNT FOR THE PERIOD OF 16-Oct-2019 TO 30-Oct-2019	2019-10-30	BDT	SAVINGS ACCOUNT	1501101613	002 01613	70515.00	78500.00	26442.39
9	STATEMENT OF ACCOUNT FOR THE PERIOD OF 15-Sep-2019 TO 13-Mar-2020	2020-03-13	BDT	SAVINGS ACCOUNT	1501101613	002 01613	835792.50	836020.85	3398.24
10	STATEMENT OF ACCOUNT FOR THE PERIOD OF 11-Feb-2020 TO 09-Aug-2020	2020-08-09	BDT	SAVINGS ACCOUNT	1501101613	002 01613	910042.29	919150.00	17685.95
11	STATEMENT OF ACCOUNT FOR THE PERIOD OF 10-May-2020 TO 09-Aug-2020	2020-08-09	BDT	SAVINGS ACCOUNT	1501101613	002 01613	554102.00	549000.00	17685.95
12	STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 30-Jun-2020	2020-10-01	BDT	SAVINGS ACCOUNT	1501101613	002 01613	1611268.79	1654790.85	61757.95

Figure 4.2: Statement Summary info in database table

Moreover, in statement details info table, we have also found in total 1146 records of 12 statements along with summary reference id. In the pdf statement, mentioned in figure 4.3, few particulars have been split into next line and date is missing in balance forward line. However, in the database table, date of balance forward line has been filled by statement start date and particulars have been picked up perfectly which depicts in figure 4.4.

STATEMENT OF ACCOUNT FOR THE PERIOD OF 01-Jul-2019 TO 22-Aug-2019

DATE	PARTICULARS	CHQ. NO	WITHDRAW	DEPOSIT	BALANCE
	Balance Forward		0.00	0.00	18,235.89
01-Jul-2019	HF YRLY AC MNT FEE-2019		100.00	0.00	18,135.89
01-Jul-2019	VAT ON HF YRLYAC MNT FEE-2019		15.00	0.00	18,120.89
02-Jul-2019	CWDR/Anik Tower ATM DHAKA BD /1501101613002		3,500.00	0.00	14,620.89
07-Jul-2019	CWDR/HAZICAMP DHAKA BD /1501101613002		10,000.00	0.00	4,620.89
07-Jul-2019	CWDR/HAZICAMP DHAKA BD /1501101613002		2,500.00	0.00	2,120.89
07-Jul-2019	IB/EFT/018662/OC/NBL/10670016575 09		2,000.00	0.00	120.89
07-Jul-2019	IB/024176/self		0.00	30,000.00	30,120.89
07-Jul-2019	CWDR/DUTCH-BANGLA BANK LTD. DHAK/1501101613623002		16,515.00	0.00	13,605.89
09-Jul-2019	CWDR/Anik Tower ATM DHAKA BD /1501101613002		4,500.00	0.00	9,105.89
11-Jul-2019	INH/006774/BKS/OT/01734479980		3,001.00	0.00	6,104.89
11-Jul-2019	CWDR/Anik Tower ATM DHAKA BD /1501101613002		5,500.00	0.00	604.89
14-Jul-2019	CWDR/Anik Tower-2nd DHAKA BD /1501101613002		500.00	0.00	104.89
15-Jul-2019	IB/028339/Personal		0.00	15,000.00	15,104.89

Figure 4.3: Sample pdf statement by mentioning problem area

Results		Messages				
ID	tran_date	tran_particulars	withdraw	deposit	balance	
532	2019-07-01	Balance Forward	0	0	18236	
533	2019-07-01	HF YRLY AC MNT FEE-2019	100	0	18136	
534	2019-07-01	VAT ON HF YRLYAC MNT FEE-2019	15	0	18121	
535	2019-07-02	CWDR/Anik Tower ATM DHAKA BD /1501101613000002	3500	0	14621	
536	2019-07-07	CWDR/HAZICAMP DHAKA BD /1501101613000002	10000	0	4621	
537	2019-07-07	CWDR/HAZICAMP DHAKA BD /1501101613000002	2500	0	2121	
538	2019-07-07	IB/EFT/018662/OC/NBL/10670016575 09	2000	0	121	
539	2019-07-07	IB/024176/self	0	30000	30121	
540	2019-07-07	CWDR/DUTCH-BANGLA BANK LTD. DHAK/1501101613000002	16515	0	13606	
541	2019-07-09	CWDR/Anik Tower ATM DHAKA BD /1501101613000002	4500	0	9106	
542	2019-07-11	INH/006774/BKS/OT/01734479980	3001	0	6105	
543	2019-07-11	CWDR/Anik Tower ATM DHAKA BD /1501101613000002	5500	0	605	
544	2019-07-14	CWDR/Anik Tower-2nd DHAKA BD /1501101613000002	500	0	105	
545	2019-07-15	IB/028339/Personal	0	15000	15105	

Figure 4.4: Details Info of sample pdf statement in database table

### 4.3 Comparative Analysis

In the related works, many sorts of off the shelf conversion libraries such as camelot, tabula, pypdf2, pdftotext which have been used as per nature of the content of the pdf document and expectation mentioned in table recognition from pdf [9], [10], [17], converting pdf to html [3], [14], extraction pdf to structured xml format [8], pdf to text extraction [24] etc. However these libraries only do the conversion and generate only raw data which contains many unexpected characters, distorted characters, space, lines, page break etc. So after conversion, they applied customize logics to eliminate unnecessary characters and properly formatted raw data. After that, based on expected result, they applied own algorithm to grab the targeted output from raw data. So, formatting technique and algorithm will be varied based on the content of the pdf document.

However, in this experiment, pdf bank account statement of BRAC bank has been used and the content and format of this pdf document is different with other documents. Besides, the content and format of account statement of different organization can be different. At first, we extracted pdf statement to text by using python library pdftotext[25] and then, applied own logics to do correction raw text and finally applied own algorithm, mentioned in section 3.6, to grab statement summary and details information. Therefore, after result analysis, we found all extracted data grabbed and stored into a SQL database properly.

# Chapter 5

## Conclusion and Future Work

### 5.1 Conclusion

Bank Account Statement (.pdf) extraction through automation plays a pivotal role in the financial sector. With the help of this automation, credit approval decision can be taken more accurately and quickly. So, manual effort can be reduced significantly as well as more credit proposal can be processed. As a result, it helps to earn more revenue and reduce non-performing loan for the organization

### 5.2 Future Work

In future work we can attempt to extract different types of bank account statement of different organizations and make it a central tool which is capable to process any organization's statement. Since, this model mainly focuses to extract pdf and store into a database, in future an analytical tool can be developed which will present analytical reports in different dimensions such as source wise total deposit graph during the statement period, withdrawal report or graph based on different purpose etc.

# Bibliography

- [1] T. Bienz, R. Cohn, and C. Adobe Systems (Mountain View, *Portable document format reference manual*. Citeseer, 1993.
- [2] F. Provost, “Machine learning from imbalanced data sets 101,” in *Proceedings of the AAAI’2000 workshop on imbalanced data sets*, AAAI Press, vol. 68, 2000, pp. 1–3.
- [3] F. Rahman and H. Alam, “Conversion of pdf documents into html: A case study of document image analysis,” in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, IEEE, vol. 1, 2003, pp. 87–91.
- [4] D. Bainbridge, K. J. Don, G. R. Buchanan, I. H. Witten, S. Jones, M. Jones, and M. I. Barr, “Dynamic digital library construction and configuration,” in *International Conference on Theory and Practice of Digital Libraries*, Springer, 2004, pp. 1–13.
- [5] I. H. Witten, K. J. Don, M. Dewsnip, and V. Tablan, “Text mining in a digital library,” 2004.
- [6] M. Ikonomakis, S. Kotsiantis, and V. Tampakas, “Text classification using machine learning techniques.,” *WSEAS transactions on computers*, vol. 4, no. 8, pp. 966–974, 2005.
- [7] Y. Ishitani, K. Fume, and K. Sumita, “Table structure analysis based on cell classification and cell modification for xml document transformation,” in *Eighth International Conference on Document Analysis and Recognition (ICDAR’05)*, IEEE, 2005, pp. 1247–1252.
- [8] H. Déjean and J.-L. Meunier, “A system for converting pdf documents into structured xml format,” in *International Workshop on Document Analysis Systems*, Springer, 2006, pp. 129–140.
- [9] S. Mandal, S. Chowdhury, A. K. Das, and B. Chanda, “Detection and segmentation of table of contents and index pages from document images,” in *Second International Conference on Document Image Analysis for Libraries (DIAL’06)*, IEEE, 2006, 12–pp.
- [10] T. Hassan and R. Baumgartner, “Table recognition and understanding from pdf files,” in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, IEEE, vol. 2, 2007, pp. 1143–1147.
- [11] Y. Liu, K. Bai, P. Mitra, and C. Giles, “Searching for tables in digital documents,” in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, IEEE, vol. 2, 2007, pp. 934–938.
- [12] G. Van Rossum *et al.*, “Python programming language.,” in *USENIX annual technical conference*, vol. 41, 2007, p. 36.

- [13] Y. Li, R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. Jagadish, “Regular expression learning for information extraction,” in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 21–30.
- [14] D. Jiang and X. Yang, “Converting pdf to html approach based on text detection,” in *Proceedings of the 2nd international conference on interaction sciences: Information technology, culture and human*, 2009, pp. 982–985.
- [15] S. Marinai, “Metadata extraction from pdf papers for digital library ingest,” in *2009 10th International conference on document analysis and recognition*, IEEE, 2009, pp. 251–255.
- [16] E. Oro and M. Ruffolo, “Trex: An approach for recognizing and extracting tables from pdf documents,” in *2009 10th International Conference on Document Analysis and Recognition*, IEEE, 2009, pp. 906–910.
- [17] ———, “Trex: An approach for recognizing and extracting tables from pdf documents,” in *2009 10th International Conference on Document Analysis and Recognition*, IEEE, 2009, pp. 906–910.
- [18] W. Wagner, “Steven bird, ewan klein and edward loper: Natural language processing with python, analyzing text with the natural language toolkit,” *Language Resources and Evaluation*, vol. 44, no. 4, pp. 421–424, 2010.
- [19] M. P. Eve, “Metadata handling for open access journal pdfs,” *martineve.com*, 2012.
- [20] K. Goslin and M. Hofmann, “Cross domain assessment of document to html conversion tools to quantify text and structural loss during document analysis,” in *2013 European Intelligence and Security Informatics Conference*, IEEE, 2013, pp. 100–105.
- [21] A. S. Alshatti, “The effect of credit risk management on financial performance of the jordanian commercial banks,” *Investment management and financial innovations*, no. 12, N<sup>o</sup> 1 (contin. 2), pp. 338–345, 2015.
- [22] S. Khusro, A. Latif, and I. Ullah, “On methods and tools of table detection, extraction and annotation in pdf documents,” *Journal of Information Science*, vol. 41, no. 1, pp. 41–57, 2015.
- [23] M. O. Perez-Arriaga, T. Estrada, and S. Abad-Mota, “Tao: System for table detection and extraction from pdf documents,” in *The Twenty-Ninth International Flairs Conference*, 2016.
- [24] H. Bast and C. Korzen, “A benchmark and evaluation for text extraction from pdf,” in *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, IEEE, 2017, pp. 1–10.
- [25] J. A. Palmer, “Pdftotext,” *Retrieved May*, vol. 25, p. 2020, 2020.
- [26] U. Kasi, “Extraction of bank transaction data and classification using naive bayes,”