

Bangor University

DOCTOR OF PHILOSOPHY

Analysing and Correcting Dyslexic Arabic Texts

Alamri, Maha

Award date:
2019

Awarding institution:
Bangor University

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 13. Mar. 2024



PRIFYSGOL
BANGOR
UNIVERSITY

School of Computer Science and Electronic Engineering
College of Environmental Sciences and Engineering

Analysing and Correcting Dyslexic Arabic Texts

Maha Marzouq Alamri

Submitted in partial satisfaction of the requirements for the
Degree of Doctor of Philosophy
in Computer Science

Acknowledgements

Always and forever, all praise and thanks be to God (Allah) for everything and for guiding me towards the successful accomplishment of my PhD—a dream that has now become a reality.

I owe many people my sincere gratitude for helping me throughout the process. I should start with expressing my special gratitude to my supervisor, Dr. William J. Teahan, for every single act of support that he has offered for me—thank you for your substantial feedback and dedicated encouragement.

I am especially and profoundly grateful to the provenience of my love, my extended family. Many special thanks and love go to my parents, Marzouq and Faddah, for their infinite love, uncountable prayers and permanent wishes. My brothers and sisters, you know how excited I was throughout the past years in spite of the anticipated obstacles, and you were such huge supporters behind the scene, making me stronger and laugh louder whilst also inspiring me to keep going. I love and thank you all for everything you have done for me.

My special gratitude and appreciation also go directly to my immediate family. My husband, Turki, I know how proud you are feeling right now and

I know it is because my dream has now been achieved. I know there are no words that can describe the uncertainty we had experienced during the long journey of my study, however, you stood by me in everything and always allowed me to nurture my ambition and sustain my enthusiasm towards this moment. My deepest thanks should go to you, and to our daughter, Deema, the light of my life, whose presence with us is the true platform to our happiness.

I would also like to give special thanks to those who participated in this research and contributed to the study in spite of their exhausting academic schedules. Your contributions are key to the successful completion of this thesis, and I am sincerely grateful for your involvement. In addition, I wish to thank everyone who offered me invaluable advice, encouragement and support, including Dukhnah Alamri (lecturer) and Asma Aljathlan, Elham Alsaleh (teachers) and Maha Alammar.

I would like to thank Bangor University for providing an ideal environment for study and research. I am profoundly thankful to all staff and fellow researchers at the School of Computer Science and Electronic Engineering for their help and direction, especially Prof. Ludmila Kuncheva, Dr. Mohammed Mabrook, Dr. Noor Al-Kazaz and Dr. Nadim Ahmed.

Finally, I would like to acknowledge my country, the Kingdom of Saudi Arabia, and Al-Baha University for providing me the invaluable opportunity to study in the United Kingdom.

I dedicate this thesis

To my parents, for their love, prayers, endless support and encouragement;

To my sister Wadha, for standing by me when things turn bleak; and

To people with dyslexia, for inspiring me.

Abstract

Dyslexia is a disorder that involves difficulty with literacy skills and language related skills. It is related to the inability of a person to master the utilisation of written language and affects a significant number of people. This thesis describes the development of the Bangor Dyslexia Arabic Corpus (BDAC) in order to facilitate the analysis and automatic correction of dyslexic Arabic text. This thesis has also developed a new classification of errors made in Arabic by people with dyslexia which was used in the annotation of the BDAC. The dyslexic error classification scheme for Arabic texts (DECA) comprises a list of dyslexia spelling errors classified into 37 types, and grouped into nine categories.

This thesis also investigates a new type of classification – dyslexia text classification – that identifies whether or not a text has been written by a person with dyslexia. The text compression scheme known as prediction by partial matching (PPM) has been applied to the problem of distinguishing dyslexic text from non-dyslexic text. Experimental results show that the F_1 score for PPM-based classification was 0.99 and outperformed other classifiers such as Multinomial Naïve Bayes and Support Vector Machines.

A new system called Sahah is also proposed for the automatic detection and correction of dyslexia errors in Arabic text. The system uses a language model based on the PPM text compression scheme in addition to edit operations (omission, addition, substitution and transposition). The correct alternative for each error word is chosen on the basis of the compression codelength. Two experiments were carried out to evaluate the usefulness of the Sahah system. Firstly, its accuracy was evaluated using the BDAC

containing errors made by people with dyslexia. Secondly, the results of Sahah were compared with the results obtained when using word processing software and the Farasa tool. The results show that the Sahah system significantly outperforms Microsoft Word, Ayaspell and the Farasa tool with an F_1 score of 0.83 for detection and an F_1 score of 0.58 for correction.

Contents

1	Introduction	1
1.1	Background and Motivation	1
1.2	Research Questions	3
1.3	Aim and Objectives	4
1.4	Contributions	5
1.5	Publications	6
1.6	Thesis Outline	8
2	Literature Review	11
2.1	Introduction	11
2.2	Dyslexia	12
2.2.1	Dyslexia Identification	13
2.3	Arabic Language	15
2.3.1	Arabic Script	15
2.3.2	Arabic Vowels	17
2.3.3	Arabic Morphology and Phonology	19
2.3.4	Arabic Encoding Methods	21
2.4	Spelling Errors	21
2.4.1	Spelling Errors by Writers with Dyslexia	23
2.4.2	Spelling Errors by Arabic Writers with Dyslexia	24

2.5	Corpus Linguistics	27
2.5.1	Types of Corpora	27
2.5.2	Dyslexia Corpora	29
2.6	Natural Language Processing	30
2.6.1	Evaluation Techniques	31
2.6.2	Text Classification	33
2.6.2.1	Multinomial Naïve Bayes	33
2.6.2.2	Support Vector Machines	34
2.6.2.3	Arabic Text Classification	35
2.6.3	Dyslexia Classification	36
2.6.4	Spelling Correction	37
2.6.4.1	The Noisy Channel Model	38
2.6.4.2	n-gram Language Models	39
2.6.4.3	Arabic Spelling Correction	41
2.6.4.4	Dyslexia Spelling Correction	43
2.7	Text Compression	44
2.7.1	Prediction by Partial Matching	45
2.7.1.1	Variants of Prediction by Partial Matching	46
2.7.1.2	Compression Codelengths using PPM-Based Models	50
2.7.1.3	Text Classification Using PPM	51
2.7.1.4	PPM Compression Method for Arabic	52
2.8	Conclusion	54
3	Bangor Dyslexic Arabic Corpus	55
3.1	Introduction	55
3.2	Identifying Dyslexia in Saudi Arabian Schools	56
3.3	Data Collection Procedure	57

3.4	Transcribing the Handwritten Data	58
3.5	The Bangor Dyslexia Arabic Corpus	62
3.5.1	Text Resources of the BDAC Corpus	63
3.5.2	Categorisation	64
3.6	Participant Information	64
3.7	BDAC Frequency Profiling	66
3.8	Conclusion	67
4	Error Annotation for Dyslexic texts in Arabic	70
4.1	Introduction	70
4.2	Basis of Dyslexic Error Classification Scheme for Arabic Texts	71
4.3	Dyslexic Error Classification Scheme for Arabic Texts	74
4.4	Evaluating the DECA	77
4.4.1	Annotation Sample Evaluation	77
4.4.2	Teachers' Feedback Evaluation	80
4.4.3	Inter-annotator Agreement Evaluation	80
4.5	The Annotation Process of the BDAC Corpus	81
4.6	Analysis of Dyslexic Errors in the BDAC	84
4.7	Conclusion	87
5	Distinguishing Dyslexic Text from Non-dyslexic Text	89
5.1	Introduction	89
5.1.1	Example of Dyslexia Classification	90
5.2	Text Corpora	91
5.3	Classification Experiments	92
5.3.1	Dyslexia Corpus and Learner Corpus Experiment . . .	93
5.3.2	Arabic Children Corpora Experiment	95
5.3.3	Experiment to Compare PPM with Other Classifiers .	100

5.4	Conclusion	102
6	Automatic Correction of Arabic Dyslexic Text	103
6.1	Introduction	103
6.2	Spelling Correction Functions	104
6.3	The Sahah System for the Automatic Spelling Correction of Dyslexic Arabic Text	106
6.3.1	Pre-processing Stage	107
6.3.2	Error Detection and Correction Stage	110
6.3.2.1	Sub-stage 2a: PPM Correction	111
6.3.2.2	Sub-stage 2b: Error Detection	115
6.3.2.3	Sub-stage 2c: Edit Operations	116
6.3.3	Post-processing Stage	117
6.4	Evaluation	118
6.4.1	Evaluation Methodology	118
6.4.2	Experimental Results	121
6.5	Conclusion	125
7	Conclusion	126
7.1	Introduction	126
7.2	Summary of the Thesis	126
7.3	Review of Research Questions	132
7.4	Review of Aim and Objectives	133
7.5	Future Work	135
	References	137
	Appendices	159

List of Figures

3.1	Screenshot of the text transcription form.	60
3.2	Example of a handwritten text written by a girl with dyslexia with the intended text transcribed by a teacher.	61
3.3	Example of a handwritten text written by a girl with dyslexia.	64
3.4	Number of documents from each category.	65
3.5	Age range of the participants.	65
3.6	Character frequency distribution.	68
4.1	Interface of the tool developed to facilitate the annotation process.	83
4.2	XML sample with stand-off annotation by tokens.	85
5.1	Dyslexia classification using PPMD.	90
5.2	Accuracy of dyslexia (BDAC) versus non-dyslexia (ALC) text classification using different PPMD orders.	94
5.3	Pre-processing and post-processing for PPM using bi-graph replacement.	97
5.4	Accuracy of dyslexia (BDAC) versus non-dyslexia (BNDAC) text classification using different BS-PPM orders.	99
6.1	Workflow of the Sahah system.	107

6.2	Workflow of the error detection and correction stage of the	
	Sahah system.	110

List of Tables

1.1	Publications that relate to this study.	7
2.1	Arabic joining groups and group letters.	16
2.2	Table of Arabic diacritics with their ALA-LC Romanization representations and samples with the character ‘ت’.	18
2.3	Confusion matrix of two classes.	32
2.4	The PPMD model after the string ‘ <i>dyslexicornotdyslexic</i> ’ has been processed.	48
2.5	Models for predicting character streams (Teahan, 1998). The symbol \hookrightarrow in the table represents an escape	51
2.6	The PPMD model after the string ‘المعلم’ has been processed. .	53
3.1	Examples of words for which it was not possible to transcribe.	62
3.2	Word frequency distribution.	66
4.1	Error types recorded in Arabic studies.	72
4.2	Version 1 of Dyslexic Error Classification Scheme for Arabic Texts (DECA).	76
4.3	Version 2 of Dyslexic Error Classification Scheme for Arabic Texts (DECA).	79
5.1	Text corpora used in the experiments.	91

5.2	Some examples of text found in each of the corpora used in the experiments.	92
5.3	Classification results of dyslexia (BDAC) versus non-dyslexia (ALC) text using different PPMD orders.	94
5.4	Classification results of dyslexia (BDAC) versus non-dyslexia (BNDAC) text using different PPMD orders.	96
5.5	Bi-graphs frequency statistics for the BDAC corpus and BNDAC corpus.	98
5.6	Classification results of dyslexia (BDAC) versus non-dyslexia (BNDAC) text using different BS-PPM orders.	99
5.7	Different order of BS-PPM over dyslexia and non-dyslexia corpora.	100
5.8	Summary of F_1 experimental results for dyslexia classification using cross-validation.	101
6.1	The most common prefixes and suffixes based on the dyslexia corpus analysis.	108
6.2	Cases of removing the redundant characters.	109
6.3	Example of an error that corrects with a different ordering of sub-stages.	111
6.4	Improving the model of sub-stage 2a.	113
6.5	Improving the model of sub-stage 2c.	113
6.6	Set of transformations rules from the DECA used for sub-stage (2a) in the Sahah system.	114
6.7	The codelength of possible alternatives spellings by using the confusions in Table 6.6 for the erroneous word “احمد”.	115
6.8	Codelengths for different candidate trigrams for a sample correction.	117

6.9	Detection and correction results after the pre-processing stage and sub-stage 2a of the Sahah system.	121
6.10	Detection and correction result after all stages and sub-stages of the Sahah system were applied.	121
6.11	Detection comparison using the Bangor Dyslexia Arabic Cor- pus (BDAC) corpus.	122
6.12	Correction comparing the Sahah system with the Farasa tool.	123

Chapter 1

Introduction

1.1 Background and Motivation

The word dyslexia originates from the Greek language and signifies difficulty with words (Ghazaleh, 2011), and specifically issues with reading, spelling, and word recognition (Grigorenko, 2001). The earliest consideration of dyslexia was presented by W. Pringle Morgan, in November 1896 (Morgan, 1896). The article described the case of a 14-year-old boy who was apparently at an adequate level of intelligence and logical reasoning for his age, yet struggled considerably in terms of reading and writing skills. This article is one of the first reports concerning congenital word blindness. Therefore, Morgan is often considered as being the pioneer in the field of dyslexia (Guardiola, 2001).

There seems to also be a significant amount of people who have dyslexia, as the International Dyslexia Association (2012) reported that dyslexia affects 15-20% of any given population. It should also be noted that there is no relationship between dyslexia and a person's level of intelligence. Dyslexia

is popularly identified with numerous famous figures, such as Richard Branson.

Dyslexia concerns difficulty with acquiring literacy skills, and this difficulty can be present throughout person's life, and might influence their education over the long-term. Furthermore, the disorder is not culture- or language-specific, and is therefore not exclusive to specific cultures, and is observable in all age groups and in all languages (Reid, 2010), such as Arabic.

The Arabic language is one of the most widely used in many parts of the world. A study conducted by Holes (2004) suggested that two fundamental reasons exist for the wide usage of this language, the first that Arabic is the language of the 'Holy Quran', and the second that other languages, such as Urdu and Farsi, employ Arabic letters. Thus, Arabic was selected as the focus of this thesis in addition to it being the researcher's native language. Moreover, the researcher wished to help Arabic people with dyslexia, and to add value for this target group.

Nevertheless, despite the widespread use of the language, academic research concerning dyslexia in Arabic is scarce, because dyslexia is not widely recognised in the Arab region (Aboudan et al., 2011). This is evidenced by the fact that the first dyslexia association in the Arab world was established in Kuwait in 1999, many years after the equivalent association was established in the West, where the oldest such association, the International Dyslexia Association was established in 1949.

There are a number of different approaches that can be employed to help and support people with dyslexia, such as tools to assist in its diagnosis and assessment, and applications such as word prediction software, text classification, and spelling correction. These tools have been developed through

different methods such as natural language processing and corpus linguistics.

The use of text corpora has expanded in recent years as it plays a significant role in different aspects such as computational linguistics, and Natural Language Processing (NLP) research. Although the use of text corpora has enjoyed a relatively high level of interest in research, availability of dyslexia corpora is scarce (Pedler, 2007), and only a few studies have considered the potential benefits of using dyslexia corpora.

Furthermore, there is an obvious lack of Arabic dyslexia corpora, which highlights the importance of improving and enlarging the extant resource that was previously developed by the researcher of this thesis (Alamri, 2013). Such a corpus can be used as a starting point for developing a more extensive understanding of dyslexic errors in Arabic, and how they are written and moreover, towards investigating and developing Arabic dyslexia applications. Furthermore, it can serve as a platform for future researchers to develop further studies in the area, or to employ in the creation of applications for dyslexics.

These points formed the main inspiration for conducting this research study.

1.2 Research Questions

The research questions explored for this study are as follows:

1. What is an effective spelling error classification scheme for annotating and analysing Arabic dyslexic corpora?
2. How well dose a compression-based language modelling method, such

as the Prediction by Partial Matching (PPM) text compression method, compare to two well performed algorithms such as Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM) for classifying a text that has been written by a person with dyslexia?

3. Can PPM, in conjunction with other methods, be effectively applied to correcting a text that has been written by a person with dyslexia?

1.3 Aim and Objectives

The aim of this study is to investigate the effectiveness of a new approach to classifying and correcting Arabic dyslexic text, specifically, using the PPM compression method. This study seeks to evaluate how well this approach performs in applications using Arabic dyslexic corpus.

Therefore, this study's objectives in investigating the research questions are as follows:

- Review the extant literature regarding dyslexia, Arabic language, dyslexia spelling errors, corpus linguistics, text classification, spelling correction, and text compression (see Chapter 2);
- Improve the existing Arabic corpus of texts written by people with dyslexia (the Bangor Dyslexia Arabic Corpus (BDAC)) (see Chapter 3);
- Create a new dyslexic error classification scheme for Arabic dyslexic texts (DECA) (see Chapter 4);
- Develop and evaluate a method to classify whether or not a text has been written by a person with dyslexia, using the PPM compression

scheme, and compare the performance of the PPM with other classification methods, such as the Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM), when they are employed for the purpose of classifying dyslexic text (see Chapter 5);

- Design and evaluate an automatic spelling correction system for correcting spelling errors in Arabic texts, produced by people with dyslexia, by comparing them with other spelling correction tools (see Chapter 6).

1.4 Contributions

Since there is currently a lack of an Arabic dyslexia corpus, this study has the potential to make valuable contributions to the field. The specific contributions are as follows:

- The first and foremost contribution of this research study is the enlargement of the Arabic dyslexia corpus (the BDAC) to comprise 28,203 words written by both male and female with dyslexia aged between 8 to 13 year olds. Based on the literature review, the BDAC is the first dyslexia corpus for Arabic.
- The second contribution is the development of dyslexic error classification scheme for Arabic texts (DECA) that can provide a framework to help analysing and annotating specific errors committed by writers with dyslexia. Also, this has been used to provide an annotated dyslexic corpus (the BDAC) and then analysis of Arabic dyslexic errors, based on the corpus.
- The third contribution is the creation of Bangor Non-Dyslexia Arabic

Corpus (BNDAC), consisting of 9,099 words written by non-dyslexic male and female between the ages of 8 and 13.

- The fourth contribution is the investigation of an effective, new method for classifying dyslexic text, based on the PPM compression method.
- The final contribution is the development and testing of a new system called Sahah to automatically correct Arabic dyslexic text by using PPM text compression scheme and an edit operation approach using compression code length.

1.5 Publications

The researcher has already published one conference paper based on this study. In addition, a further two journal papers have been submitted for publication. Table 1.1 shows specific papers which relate to this study.

The first paper, entitled “A New Error Annotation for Dyslexic Texts in Arabic”, is included in Chapter 4. The paper describes a new classification scheme of errors made in Arabic by people with dyslexia to be used in the annotation of the Arabic dyslexia corpus (BDAC). The dyslexic error classification scheme for Arabic texts (DECA) comprises a list of spelling errors extracted from previous studies and a collection of texts written by people with dyslexia that can provide a framework to help analyse specific errors committed by writers with dyslexia. The classification comprises 37 types of errors, grouped into nine categories. The paper also discusses building a corpus of dyslexic Arabic texts that uses the error annotation scheme and provides an analysis of the errors that were found in the corpus. The paper was presented at the Third Arabic Natural Language Processing Workshop

Table 1.1: Publications that relate to this study.

1	Title	A New Error Annotation for Dyslexic Texts in Arabic
	Authors	Maha M. Alamri, and William J. Teahan
	In	The Third Arabic Natural Language Processing Workshop (WANLP)
	Publisher	Association for Computational Linguistics (ACL)
	Year	2017
	Status	Published
2	Title	Distinguishing Dyslexic Text from Non-dyslexic Text
	Authors	Maha M. Alamri, and William J. Teahan
	In	Transactions on Computers (TC) Journal
	Publisher	IEEE
	Year	2019
	Status	Submitted
3	Title	Automatic Correction of Arabic Dyslexic Text
	Authors	Maha M. Alamri, and William J. Teahan
	In	Computers Journal
	Publisher	Multidisciplinary Digital Publishing Institute (MDPI)
	Year	2019
	Status	Published

(WANLP) co-located with EACL 2017, held in Valencia, Spain.

The second paper, entitled “Distinguishing Dyslexic Text from Non-dyslexic Text”, which Chapter 5 is based upon, investigates a classification problem, specifically dyslexia text classification, which involves identifying whether or not a text has been written by a person with dyslexia. For this purpose, we apply the PPM text compression scheme for the binary classification problem of distinguishing dyslexic text from non-dyslexic text. Various experiments were conducted to evaluate the method using three corpora. Experimental results show that the accuracy for PPM-based classification significantly outperformed standard feature-based classifiers such as Multi-

nomial Naïve Bayes (MNB) and Support Vector Machines (SVM). The paper has been submitted to the Transactions on Computers (TC) Journal (IEEE).

The third paper, entitled “Automatic Correction of Arabic Dyslexic Text”, is included in Chapter 6. This paper proposes an automatic correction system that detects and corrects dyslexia errors in Arabic text. The approach uses a language model based on the PPM text compression scheme that generates possible alternatives for each error word. Furthermore, the generated candidate list is based on edit operations (omission, addition, substitution and transposition) and the correct alternative for each error word is chosen on the basis of the compression codelength. The system is compared with widely used Arabic word processing software and the Farasa tool. The approach provided good results compared with the other tools. The paper was published in the Computers Journal. Multidisciplinary Digital Publishing Institute (MDPI).

1.6 Thesis Outline

This thesis is organised into seven chapters. The outline of each chapter is as follows:

This chapter presented the background and motivation of this research study, the aim and objectives and the research questions. It also lists the contributions of the study to the field of Arabic dyslexia studies, and noted the papers based on the study that have already been published or submitted.

Chapter 2 provides a review of the extant literature of relevance to the sub-

ject of this thesis, including an overview of the previous studies that focused on various associated issues as the main area of interest of this study is the zone of convergence between several aspects, including dyslexia, the Arabic language, corpora linguistic and NLP. This chapter comprises a review of the literature concerning dyslexia followed by the Arabic language and its specific linguistic characteristics. Then, spelling errors and more specifically dyslexia errors are discussed. Followed by a review of the literature concerning corpus linguistics and types of corpora. After that, the literature concerning two NLP tasks –text classification and spelling correction– are discussed. This is followed by an introduction to text compression and the PPM text compression method is introduced in detail. Moreover, how the PPM compression method can be adapted for the Arabic language is described.

Chapter 3 describes how the BDAC corpus was compiled and how dyslexia is identified in schools in Saudi Arabia, together with how the handwritten data was converted into an electronic format, and analyses of the text and participant information.

Chapter 4 provides a detailed description of the basis of the new dyslexic error classification scheme for Arabic dyslexic text (DECA), and evaluates DECA and how the consistency between annotators using the classification scheme can be measured. It also explains the annotation process of the BDAC corpus and provides an analysis of Arabic dyslexic errors.

Chapter 5 investigates a compression-based classification method employed to distinguish dyslexic text from non-dyslexic text, using three corpora, and also discusses the experiments conducted for the purpose of comparing the PPM with other classification methods (Multinomial Naïve Bayes and

Support Vector Machines).

Chapter 6 discusses the development of an Arabic automatic spelling correction system, the Sahah system, which includes a number of different stages. It also evaluates this system, and compares it with other widely used spellchecking software and the Farasa tool.

Finally, Chapter 7 summarises the study, and provides the overall results and their significance in relation to the research questions, and the recommendations for future research.

Chapter 2

Literature Review

2.1 Introduction

This chapter examines the background technologies in this study and explains the concept of the theoretical framework for dyslexia, Arabic language, spelling errors, corpus linguistics, text classification and spelling correction and ends with Prediction by Partial Matching (PPM) compression scheme.

This chapter will firstly gives an overview and definitions of dyslexia in Section 2.2. Section 2.3 discusses the Arabic writing script, Arabic vowels and Arabic morphology and phonology. The spelling errors of dyslexic writers are discussed in Section 2.4. Section 2.5 discusses corpus definitions and types in addition to dyslexia corpora. After that, Section 2.6 review natural language processing applications involving text classification in Section 2.6.2 and discusses various techniques in addition to Arabic text classification and dyslexia classification. This is followed by a discussion of spelling correction in Section 2.6.4 including Arabic spelling correction in addition to dyslexia

spelling correction. Section 2.7 discusses the fundamentals of text compression and Prediction by Partial Matching (PPM) in Section 2.7.1.

2.2 Dyslexia

Dyslexia encompasses a wide array of learning difficulties. Thus, dyslexia is sometimes called ‘the mother’ of learning difficulties (Davis and Braun, 1997). The word dyslexia has its origin in Greek. It is comprised of ‘dys-’, which means difficulty with, and ‘-lexia’, which means language or words (Ghazaleh, 2011).

Dyslexia is defined by the International Dyslexia Association (2002) as a neurobiological condition characterised by an individual’s inability to read, spell, decode text and recognise words accurately or fluently. The British Dyslexia Association (2007) defines dyslexia as follows: “Dyslexia is a specific learning difficulty that mainly affects the development of literacy and language related skills. It is likely to be present at birth and to be life-long in its effects. It is characterised by difficulties with phonological processing, rapid naming, working memory, processing speed, and the automatic development of skills that may not match up to an individual’s other cognitive abilities”. According to Mortimore (2008), dyslexia arises due to a deficiency in the phonological dimension of language. In addition, dyslexia impedes the improvement of key language skills, including reading, writing and spelling.

Dyslexia is observable across different languages (Elbeheri and Everatt, 2007). The manifestation of dyslexia may vary across languages, since languages vary in the way in which their orthography represents phonol-

ogy (Reid, 2010). As such, the severity of reading, writing and spelling deficits vary across different language orthographies (Elbeheri et al., 2006). Readers in languages with a transparent orthography, such as Spanish, face fewer difficulties than readers in languages with non-transparent orthography such as English (Rello, 2014).

According to a number of studies, there are genetic and hereditary factors that determine whether dyslexia is transferred from one generation to the next. Hall (2009) asserts that there is a 50% chance that a child from a family with a history of dyslexia will develop this mental illness. A study by Vellutino et al. (2004) provides additional support for the notion that the development of dyslexia in children can be heavily influenced by genetic factors revolving around existing cognitive deficits.

Besides genetic factors, environmental factors are also responsible for weakening the cognitive skills of children that are directly linked with the occurrence of dyslexia. Snowling (2012) and Vellutino et al. (2004) recommended that parents should therefore control environmental factors that may eventually cause dyslexia. Snowling et al. (2007) also concluded in their study that environmental factors are the most significant and essential factors involved in the prevention of dyslexia. However, it is still unclear whether the occurrence of dyslexia and the strength of its impact are conditioned by genetic or environmental factors.

2.2.1 Dyslexia Identification

For most of the existing research, the criticality of early identification and intervention is continually stressed (Prevett et al., 2013), to prevent the person with dyslexia having to go through a stressful, downward spiral

of underachievement, lowered self-esteem and poor motivation (Snowling, 2013)

Mohamad et al. (2013) emphasises that difficulties in learning, reading or writing can make children become frustrated if the dyslexic problem is undiscovered. Also, a long term effect of dyslexia is that the children may have a lack of confidence, be unmotivated or have low self-esteem.

Therefore, identification of dyslexia can ameliorate its effects since people with dyslexia can learn to cope up with their struggles and difficulties and avoid its consequences such as high rates of academic failure (Rello and Ballesteros, 2015).

There are various methods for identifying dyslexia such as, teacher observation, dyslexia checklists and interviews which are widely used for identification purposes (Alnaim, 2015). Some Ministries of Education in countries such as Malaysia and Saudi Arabia use the dyslexia checklist (Zainuddin et al., 2018; Alnaim, 2015) as the instrument to identify the probability of the children having learning difficulty specific to dyslexia which measures their capability in spelling, reading, and writing. However, there is no one ideal or agreed identification method (Alnaim, 2015).

Efforts to identify dyslexia are truly multidisciplinary and it is not specific to the field of education. More detail about methods and techniques in the fields of neuroimaging and computer science are described in Section 2.6.3.

This thesis focuses on Arabic dyslexia text specifically. Thus, the following section describes the relevant fundamentals of the Arabic language.

2.3 Arabic Language

“You read Arabic with your soul first, then with your eyes.”

— el Seed, Artist

The Arabic language “اللغة العربية” is the fourth largest language group in the world, spoken by 315 million people in 58 countries (Simons and Fennig, 2018). It is the first language of the Arab world (i.e. Egypt, Tunisia and Saudi Arabia and others). While Arabic is the primary language of the Arab world, it is also widely spoken in many non-Arab nations, such as the Central African Republic of Chad (Comrie, 2009).

2.3.1 Arabic Script

In Arabic, there are 36 Arabic letters, which can be classified as follows: the basic letters consisting of 28 letters: ا ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق and six Hamza letters, ء إ أ ؤ ئ, which are not formally included in the alphabet. There is also a hybrid letter Tah Marbuta, which merges the letters Hah ‘ه’ and Tah ‘ت’. Similarly, the hybrid letter Alif Maksura combines the letters Alif ‘ا’ and Ya ‘ي’ (Habash, 2010).

Reading and writing occur from right to left. Most letters are written in a cursive fashion (Elbeheri et al., 2006). The majority of these letters can be written in more than one form based on the position of a given letter in a word. In other words, the form of Arabic letters changes in accordance with their position within words, that is, whether they are placed at the beginning, in the middle or at the end of the word or are isolated (Brosh, 2015). For example, the letter ‘ت’ has three constituent shapes: at the beginning of a word it is ‘ت’; in the middle it is ‘ـتـ’; and at the end it is

‘ت’. Notably, the letters ‘ارزودد’ are only connected to the previous letter, located on their right, but are not joined to any subsequent letter on the left (Brosh, 2015).

It needs to be underlined that the Arabic alphabet has letters that can be grouped depending on their basic letter shape. Several letters can share the same shape, using dots under or above this shape, and the different number of dots provides a necessary means of distinguishing between these letters. More specifically, 15 letters have dots, with 10 having one dot, three having two dots and two having three dots (Abu-Rabia and Awwad, 2004). A good example of this is the shape ‘ح’, which serves as a basic shape for three Arabic letters ‘ح ح ح’. Table 2.1 provides a list of all the basic shapes, with letters that are formed using a particular shape.

Table 2.1: Arabic joining groups and group letters.

Letters	Group Letters	Letters	Group Letters
ا	أ إ إ	ف	ف
ب	ب ت ث	ق	ق
ح	ح ح ح	ك	ك
د	ذ	ل	ل
ر	ر ز	م	م
س	س ش	ن	ن
ص	ص ض	ه	ه ة
ط	ط ظ	ي	ي ي ئ
ع	ع غ	ء	ء

There are two most commonly recognised forms of Arabic:

- Modern Standard Arabic, also known as literary Arabic. This form of

Arabic is governed by a different set of rules regarding its vocabulary, grammar, phonology, morphology and system than the spoken form of Arabic (Abu-Rabia, 2000). This is not to say that Modern Standard Arabic cannot be used in spoken form; however, this use is limited to more formal occasions (Abu-Rabia and Taha, 2006). This form of Arabic is a part of language curricula at schools, as a good command of Modern Standard Arabic is necessary if one wants to communicate with the rest of the Arab world on a formal level (Abu-Rabia and Sammour, 2013). Modern Standard Arabic is commonly written rather than spoken (Biadisy et al., 2009).

- Arabic dialects in some ways are completely different from Modern Standard Arabic and generally used in informal daily communication (Almeman and Lee, 2013). There are different dialects based on geography such as Gulf, Iraqi, Egyptian and Maghrebi. They are not taught in schools or even standardised. Dialects are mostly spoken, not written (Biadisy et al., 2009).

2.3.2 Arabic Vowels

As is the case with most common languages, words in Arabic are composed of a combination of consonants and vowels (both short and long) (Abu-Rabia and Taha, 2006). Diacritical marks are used in the Arabic alphabet to provide signs regarding the phonology of the Arabic language (Gutub et al., 2008). Diacritics are small symbols that are placed above or below letters, the purpose being to indicate to the reader how the short vowels in certain words should be pronounced (Al Rowais et al., 2013). It is worth recognising that short vowels are not considered to be independent letters;

rather, they are represented as additional diacritical marks. Three symbols are used to mark three short vowels (, ' and). Another type of diacritical mark, the Tanwin, is used to signal double case endings. It comprises three different cases: , , ' (Harrat et al., 2013). For example, the word “كرة” [E: “a ball” B: “krpK” R: “krtin”]¹, the “ ” [R: “in”] indicates the kasratin short vowel. Table 2.2 shows the Arabic diacritics marks.

Table 2.2: Table of Arabic diacritics with their ALA-LC Romanization representations and samples with the character ‘ت’.

Diacritic	ALA-LC Romanization	Sample
َ	fathḥ ‘a’	تَ
ُ	ḍmh ‘u’	تُ
ِ	ksrh ‘i’	تِ
ً	tanwin alfath ‘an’	تً
ٌ	tanwin alḍm ‘un’	تٌ
ٍ	tanwin alksr ‘in’	تٍ

In terms of long vowels (Almadd), there are three types in Arabic: ا, و and ي. When the text written in Arabic is fully vowelised, it contains both consonants and long and short vowels (Abu-Rabia and Sammour, 2013), and it represents an example of shallow orthography; on the other hand, an unvowelised text (i.e. a text without short vowelisation) is known as deep orthography (Abu-Rabia and Taha, 2006).

The use of diacritic marks is not very common in modern standard Arabic (Aabed et al., 2007). This use can be found in reading material for children, dictionaries and the Holy Quran (Abu-Rabia and Sammour, 2013).

¹The syntax used here is as follows: “Arabic text” [E: “English translation” B: “Buckwalter transliteration” R: “ALA-LC Romanization”].

The reason that these texts are fully vocalised stems from the fact that only a minor change in short vowelisation can have a considerable impact on the meaning of a particular word and by extension the whole text (Gutub et al., 2010). For instance, a diacritised version of “الجد” can take the form of “الجدُّ” [E: “grandfather”] or “الجدّ” [E: “hard work”]. In the unvowelised text, it is up to the reader to determine the correct diacritic from the context (Elshafei et al., 2006).

2.3.3 Arabic Morphology and Phonology

Arabic has a rich and complex morphology. Hence, words may be presented in various different forms and can be modified with the addition to the root of suffixes, prefixes or both. For example, the Arabic word “يكتبه” [E: “writing it” B: “yktbh”] can be analysed as “ي+كتب+ه” [B: “y+ktb+h”], where the root is “كتب” with one prefix ‘ي’ [B: ‘y’] and one suffix ‘ه’ [B: ‘h’]. In Arabic, the conjunction ‘و’ [E: “and” B: ‘w’] is used to connect phrases or groups of words; for example, “ومدرسة” [E: “and school” B: “w mdrsp” R: “w mdrsh”].

Furthermore, Arabic contains both masculine and feminine genders and uses singular, dual and plural forms – “هو” for masculine singular and “هي” for feminine singular, “هما” for masculine and feminine dual and “هم” and “هن” for masculine and feminine plural forms, respectively (Altantawy et al., 2010).

The Arabic language demonstrates phoneme–letters correspondence, or a reliable letter-sound correlation (Abu-Rabia, 2001). Despite this, irregularities may be detected in the pronunciation of several letters and vowels. In addition, the writing rules for many letters constitute exceptions, and in

some cases letters either have added sounds or lose their sound. These additions or omissions are frequently dependent upon the position of the letter within the word (Abu-Rabia, 2001; Brosh, 2015).

For example, the often-silent letters ا, و and ي can manifest as long vowels when located in the middle of a word. But, when two of them are placed together in the middle of a word, they demonstrate another form, such that only the second letter is lengthened. Moreover, when these letters appear at the end of words, they do not lengthen at all. Another irregularity is seen with the letter 'ج' of the definite article "ال" when it is spelled. However, its phoneme is exchanged for any one of the fourteen letters known as sun letters ت ث د ذ ر ز س ش ص ض ط ظ ل ن . Other examples of irregularities include variations in letter sounds based on their location within a word's letter sequencing. For example, the letter 'س' in the word "سوط" resembles the sound of the letter 'ص', and the letter 'ت' in the word "صوت" sounds like the letter 'ط' (Brosh, 2015).

Hamza forms are determined by complex spelling rules that reflect both the vocalic context and the surrounding letters. The Tah Marbuta 'ة' is a special morphological marker that indicates a feminine ending and only appears in the last position in words. If the morpheme it denotes lies between the first and final part of a word, it takes the written form Tah 'ت'. Similarly, the morphological marker Alif Maksura 'ى' denotes a spectrum of information, including feminine word endings and original root words. When placed in the final position in a word, it is only written as a dotless Ya 'ي', but when it is in the middle position, it is represented by the letters Alif 'أ' or Ya 'ي' (Habash, 2010).

2.3.4 Arabic Encoding Methods

Arabic characters are represented digitally using different encoding methods such as the ISO 8859-6 standard, Windows-1256 and UTF-8 encoding. 89.8% of Web pages use UTF-8 encoding which has become the predominant character encoding scheme for the World Wide Web (W3Techs, 2013). UTF-8 encoding is defined by the Unicode standard and it is a compromise encoding method. It can represent any Unicode character with one byte or more (up to four bytes). One byte, the same as the American Standard Code for Information Interchange (ASCII), is used to represent an English character. The efficiency and compatibility shown by UTF-8 encoding for both ASCII text and Unicode scripts (that need more than one byte to represent each character, such as Arabic, Chinese and Japanese) has given it preference in many applications, websites and operating systems (Alhawiti, 2014).

Having discussed dyslexia and Arabic language in detail, the following section will present common spelling errors with a specific focus on errors related to dyslexia and dyslexic errors in Arabic.

2.4 Spelling Errors

Kukich (1992) categorises three types of misspellings: typographic, cognitive and phonetic.

- Typographic errors occur when the correct and proper spelling of a word is known, but it is typed incorrectly; for example, the misspelling of “two” as “tow”.

- Cognitive errors are those that arise when a word’s correct spelling is not known by its user. There is often some comparison that can be made between a correctly spelled word and its misspelling (for example, “pain” becomes “pine”) (Gupta and Mathur, 2012).
- Phonetic errors are considered a type of cognitive error, as the user of the word will use a variation on the word that, although incorrectly spelled, makes phonetic sense (for example, “speshal” instead of “special”) (Kukich, 1992).

Damerau (1964) notes that 80–95% of misspellings take the form of a word that has a similar number of letters as the correct word would have. There are four different ways in which misspellings may occur:

1. Addition when a word includes an extra letter (e.g. “universsity” instead of “university”)
2. Omission when a word has a letter omitted (e.g. “universty” instead of “university”)
3. Substitution when a word has a letter substituted (e.g. “unaversity” instead of “university ”)
4. Transposition when two letters switch position in a word (e.g. “unviersity” instead of “university”)

These misspellings are referred to as single errors. A misspelled word with additional mistakes is referred to as a multi-error misspelling.

Another way of describing spelling errors is to classify them as non-word errors and real-word errors. A non-word error – for example, the use of “reimebber” instead of “remember” – does not have a set meaning and can-

not be located within a dictionary (Mishra and Kaur, 2013; Samanta and Chaudhuri, 2013). Real-word errors occur when a writer types an actual word when a different word was intended. This word has a meaning, but it does not fit correctly into the given sentence or give the intended meaning (Islam and Inkpen, 2009).

2.4.1 Spelling Errors by Writers with Dyslexia

Many studies have observed that people with dyslexia struggle more with words with difficult structures and spellings than with words that can be easily retained through the process of repetition or which have more simple spellings (Fischer et al., 1985; Moats, 1993). For example, words with silent letters must be learned so that the proper spelling may be used in the future. An example of this is the combination “kn”, which may be seen in words in English such as “knight” and “knife”. Another example is the word “musician”, which may be confusing to people with dyslexia due to the pronunciation of the ‘c’, which is different from that in “music”. Additionally, some words may change in spelling when affixes are included, such as “explain” to “explanation” and “miracle” to “miraculous” (Bourassa and Treiman, 2008). Manis et al. (1990) categorises people with dyslexia into three main groups: those with difficulties with phonology; those with difficulties with orthography and those who experience difficulties in both areas.

One challenge commonly faced by dyslexia is how best to deal with words that are uncommonly used and which therefore are outside of the typical vocabulary (Meyler and Breznitz, 2003). As a result, even dyslexics who normally deal well with phonological challenges across academia find it difficult

to address words that require memorisation to use appropriately (Kemp et al., 2009). This may arise from a lack of experience in reading, which results in students not encountering words in the past. It may also come from difficulties in retaining the necessary orthography, which may be due to a poor visual memory that in turn leads to an inability to recall the correct order of letters within a given word.

Writers with dyslexia may experience difficulty in differentiating between how a word sounds and how it is spelled. The issues surrounding phonetics and morphology may have an impact on their ability to adapt to a language's orthography (Korhonen, 2008). Nonetheless, the number of errors a writer with dyslexia may make is mostly affected by the type of writing system specific to the language that they are using (Lindgrén and Laine, 2011). Research into this area has concluded that dyslexia across languages and linguistic systems shares the same difficulties in phonology and orthography (Aaron, 1989; Abu-Rabia, 2001; Abu-Rabia et al., 2003). Therefore, the following section will discuss spelling errors specifically by Arabic writers with dyslexia.

2.4.2 Spelling Errors by Arabic Writers with Dyslexia

According to Goulandris (2003), the manner of dyslexia varies across languages because the orthographic system is different to those used in other languages. For instance, the Arabic language has unique characteristics, such as diacritics, while some letters are written cursively and change their forms according to their position in the word, which is not the case for other languages, such as English.

It should be pointed out that there have only been limited studies into

dyslexia in Arabic due to the fact that dyslexia is not recognised in many Arabic cultures as a particular type of reading and writing issue; hence, academic research and interest in this area has been minimal (Abu-Rabia and Taha, 2004). However, there has been a substantial effort on the part of educational figures and organisations to bring more attention to the existence of dyslexia (Elbeheri et al., 2006).

According to Ali (2011), spelling errors often cause letter reversals, also known as mirror writing and writing from left to right. As Arabic is written from right to left, writing from left to right can still result in a correctly written sentence; however, mirror writing will cause the sentence to be reversed. Ali (2011) also mentions other common errors, including omission, addition, substitution and transposition. People with dyslexia also have difficulties differentiating between letters with similar forms and different sounds.

One study by Abu-Rabia and Taha (2004) examined the spelling errors observed in Arabic writers using three types of participants: dyslexic; aged-matched readers and a young reader group (matched with the dyslexics by reading level). This revealed seven types of errors: phonetic errors; semi-phonetic errors; dysphonetic errors; students may spell an Arabic word according to how it is pronounced in the local spoken dialect of Arabic; visual letter confusion; irregular spelling rules; word omission and functional word omission.

Hamadneh et al. (2014) also studied the common errors of students with learning difficulties; however, they used the viewpoints of teachers to classify and distinguish 28 different kinds of errors including but not limited to “الخلط بين التاء والتاء المفتوحة والتاء المربوطة” [E: “Confusion in Tah, Tah Marbuta and Ha”],

[E: “N in Tanwin”] [E: “Sun Letter”], “النون اخر الكلمة : النون الأصلية والتنوين” [E: “Hada and Allty”], “الخلط بين الحركات” [E: “Confusion in short vowels and AlMadd”] and “حذف حروف من” [E: “Omission”]. In Saudi Arabia, Abunayyan (2003) created a form called “تحليل الأخطاء في مادة الإملاء” [E: “Error Analysis in Spelling”], which is used in Saudi Arabia to analyse the spelling errors of students with dyslexia in primary schools. It contains 23 different error types such as “عدم معرفته للشواذ مثل” [E: “Irregular spellings like Lakn”], “حذف اللام الشمسية” [E: “Deleting Sun Letter”], “عدم التفريق بين الألف الممدودة و الألف المقصورة” [E: “Not distinguishing between Alif and Alif Maksura”], “كتابة كلمة غير التي قيلت مشابهة لها في المعنى مثل التلميذ - الطالب” [E: “Writing a word that is similar to the meaning like pupil - student”] and “عكس الحروف” [E: “Substitution”].

In 2013, the researcher of this thesis examined the errors of students with dyslexia by creating an initial version of the Bangor Dyslexia Arabic Corpus (BDAC) which has been substantially expanded in this study (see Chapter 3). The corpus consisted of 1,067 words, and 694 errors were identified. The Arabic texts were composed by female students with dyslexia aged 8–10 years. During the analysis, the researcher identified a number of spelling errors, including: an inability to specify the correct form of the Hamza; difficulty with short and long vowels; difficulty with Tanwin; omission, addition, substitution and transposition; and exchanging ظ with ض, ض with ت, ظ with ة or ؤ, and ة or ؤ with ت.

The next section explain the concept and types of corpora in addition to dyslexia corpora more specifically.

2.5 Corpus Linguistics

The term corpus (singular form of corpora) can be defined as “an electronically stored collection of samples of naturally occurring language” (Hunston, 2006). McEnery and Wilson (2001) states that “any collection of more than one text can be called a corpus: the term corpus is simply the Latin for ‘body’, hence a corpus may be defined as any body of text”. For Bennett (2010), a corpus represents a set of elements of a language that commonly occur in the production of this language and can be utilised in the process of deep investigation of various phenomena regarding this language.

A corpus denotes a significant organised collection of texts that encompass words concerning different domains of a given language. Moreover, the corpus can be understood as a set of written or spoken records of language stored in a database in an electronic form (McCarthy, 2004).

Corpora have become essential with respect to the advancement of computational techniques that are used in the field of linguistics and also, common applications are in the field of natural language processing for example, speech recognition and optical character recognition (OCR) (AbdelRaouf et al., 2010).

2.5.1 Types of Corpora

There are various types of corpora, with the differences based mainly on the type and purpose of the collected language material. However, at the same time, most corpora share certain common aspects, such as the fact that they contain texts from the same language or a particular type of language. Besides the texts from which they are composed, it is common for corpora

to include information regarding these texts (Hunston, 2006). The following is a list of the most common types of corpus (Hunston, 2002):

Specialised corpus: The size of this type of corpus may be small or large, and it is most commonly composed for use in investigations answering a specific inquiry. An example of a specialised corpus is the CHILDES Corpus (MacWhinney, 1996), which includes children’s language.

General corpus: A general corpus contains various types of texts in an attempt to encompass a wide range of different texts. Hence, it tends to be bigger than a specialised corpus, and it is often utilised in linguistics for general purposes (e.g The British National Corpus (BNC) (Aston and Burnard, 1998)).

Comparable corpora: This is a combination of corpora from different languages or from different domains of the same language. Therefore, the main purpose of these types of corpora is to offer a platform for comparing and contrasting language. An example is the International Corpus of English (ICE) (Greenbaum and Nelson, 1996).

Parallel corpora: This type of corpora consists of several corpora in different languages, where each corpus encompasses texts translated from one language to another. Currently, parallel corpora are important for observing the nature of translation. An example would be Europarl (a parallel corpus for statistical machine translation) (Koehn, 2005)

Pedagogic corpus: This type of corpus comprises of texts produced or used in classroom settings. As such, it may consist of academic textbooks or generally of any form of language, either written or spoken, that was recorded in this setting. The purpose of this type of corpus is to explore various aspects of teacher-student interaction and to help in teacher self-development.

An example is the Pedagogic Corpora for Content and Language Integrated Learning (Kohn, 2012).

Learner corpus: This is a corpus containing an accumulation of texts created by students of a particular language, which is usually used to investigate students' errors. By utilising this corpus, it is possible to identify points where students and native speakers differ with respect to language production. A well-known learner corpus is the International Corpus of Learner English (ICLE) (Granger, 2003).

Monitor corpus: This type of corpus is created to assist in identifying developments in a language. For this reason, a monitor corpus grows continuously, as new texts are being constantly added. A well known monitor corpus is the Bank of English corpus (Järvinen, 1994).

2.5.2 Dyslexia Corpora

There is a noticeable absence of corpora designed specifically for the needs of dyslexic research. However, there are three notable studies which created dyslexic corpora as follow:

- The Real-Word ERRor (RWERR) employed by Pedler (2007). This corpus comprises approximately 12,000 English words and 833 marked-up errors. Structurally, this corpus consists of different resources, namely homework by a child with dyslexia, compositions written by school leavers in the 1960's, office documents written by workers with dyslexia, online typing texts, texts created by students with dyslexia studying for the IT NVQ, a dyslexia mailing list, essays written by university-level student with dyslexia, stories written by primary school child with dyslexia and dyslexia bulletin board.

- The Spanish corpus (Dyscorpus), created by Rello (2014), was collected from children with dyslexia aged 6-15 years. This includes 83 texts, 54 from school essays and homework exercises and 29 from parents of dyslexic children, with a total of 1,057 words. Moreover, Dyscorpus is annotated and provides a list of the unique errors.
- A German study collected texts from homework exercises, dictations and school essays; it is composed of 47 texts, with participants aged between eight and 17 years old. The texts contained a list of 1,021 errors. Furthermore, a resource of German errors was created and errors were annotated (Rauschenberger et al., 2016).

Given the above, the dyslexia corpora that have been produced are found in the Latin-based languages, but no similar corpora based on Arabic is found apart from the researcher’s previous work (Alamri, 2013). More details about the expansion of the Bangor Dyslexia Arabic Corpus (BDAC) is explained in Chapter 3.

2.6 Natural Language Processing

Natural Language Processing, is an area of research and applications that explores how computers can be used to understand and manipulate natural language text or speech. The foundations of natural language processing lie in a number of disciplines, such as linguistics and artificial intelligence. Applications of natural language processing include machine translation, speech recognition and summarisation (Chowdhury, 2003).

There are some natural language processing applications that can help people with dyslexia to mitigate their problems and to alleviate potential stress

and reduce frustrating experiences of life. Some examples of these tasks are text classification and spelling correction. Thus, the link-up between natural language processing applications and dyslexia needs could improve both their quality of life and the quality of their writing.

Therefore, this thesis focused on these two tasks. The next sections review the literature of text classification, methods used for classifying texts and dyslexia classification in particular. This is followed by Arabic spelling correction methods used for correcting texts and dyslexia spelling correction.

2.6.1 Evaluation Techniques

In order to evaluate how well a classification model or correction process performs, to measure the success of the classification/correction technique and to compare the results against other techniques, a number of evaluation method and criteria may be used as follows:

Cross-validation: This is a verification technique that evaluates the generalisation ability of a classifier/corrector on an independent dataset (Awad and Khanna, 2015). k -fold cross validation separates the set of data into k sections, each of which are utilised as the test document one-by-one. The remaining $k-1$ parts are applied for training purposes.

Confusion Matrix: This is a matrix that visualises the performance of the classification/correction algorithm using the data in the matrix. It compares the predicted classification/correction against the actual classification/correction in the form of false positives, true positives, false negatives and true negatives (Awad and Khanna, 2015). The confusion matrix from the binary problem is shown as follows in Table 2.3:

Table 2.3: Confusion matrix of two classes.

	Predicted target	Predicted Non-target
Actual target	TP	FN
Actual Non-target	FP	TN

The performance of the classifier/corrector is evaluated using Recall, Precision, F_1 score and Accuracy which are calculated as follows:

$$Recall (Rec.) = \frac{TP}{TP + FN}. \quad (2.1)$$

$$Precision (Prec.) = \frac{TP}{TP + FP}. \quad (2.2)$$

$$F_1 \text{ score} = 2 \times \frac{R \times P}{R + P}. \quad (2.3)$$

$$Accuracy (Acc.) = \frac{TP + TN}{TP + TN + FP + FN}. \quad (2.4)$$

where:

- 1) True Positive (TP): This is the number of target documents that are successfully (classified/detected or corrected) as target documents.
- 2) False Negative (FN): This is the number of target documents that are unsuccessfully (classified/detected or corrected) as non-target documents.
- 3) False Positive (FP): This is the number of non-target documents that are unsuccessfully (classified/detected or changed) as target documents.

- 4) True Negative (TN): This is the number of non-target documents that are successfully (classified/detected or unchanged) as non-target documents.

2.6.2 Text Classification

According to Zhang et al. (2008), ‘text classification’ (or text categorisation) is the act of organising documents according to a single pre-determined set of categories or classes, based on their content. The use of predefined categories is a supervised learning (machine learning) approach, as it needs training data that are already categorised for the purpose of creating models that can be utilised for categorising test data (Frank et al., 2000). There are different applications of text classification, such as authorship identification, spam filtering and dialect classification. This thesis focused on dyslexia text classification.

Text classification can be implemented using a number of methods, including Naïve Bayes (NB), Multinomial Naïve Bayes (MNB), Support Vector Machines (SVM) (Vasa, 2016) and PPM compression (Frank et al., 2000; Teahan and Harper, 2003). MNB and SVM are well-known methods and commonly used for the text classification task (Kowsari et al., 2019). Thus, these two methods are explored below in more detail. These two methods are also used for comparison purposes in Chapter 5 while PPM will be discussed later in this Chapter.

2.6.2.1 Multinomial Naïve Bayes

Multinomial Naïve Bayes estimates the conditional probability associated with a given word, term, or token, which can thus be assigned to a class by

means of the term’s relative frequency in all the associated documents (Jurafsky and Martin, 2018). The probability of a class value c given a test document d is computed as follows:

$$P(c|d) = \frac{P(c) \prod_{w \in d} P(w|c)^{n_{wd}}}{P(d)}. \quad (2.5)$$

where n_{wd} is the number of times word w occurs in document d , $P(w|c)$ is the probability of observing word w given class c , $P(c)$ is the prior probability of class c , and $P(d)$ is a constant that makes the probabilities for the different classes sum to one. $P(c)$ is estimated by the proportion of training documents pertaining to class c and $P(w|c)$. This is calculated by how many times the word occurs in the training set of document class with one added in order to initialise each word count to one instead of zero. Otherwise the $P(c|d)$ will be zero (Frank and Bouckaert, 2006) divided by the number of distinct words there are in all training documents plus how many total words in the training set.

2.6.2.2 Support Vector Machiness

Support Vector Machiness (SVMs) were introduced by Vapnik (Cortes and Vapnik, 1995) and first used for text categorisation by Joachims in 1998 and subsequently used in other text categorisation research (Drucker et al., 1999; Dumais and Chen, 2000).

Support Vector Machiness seek a decision surface to separate the training data points into two classes and make decisions based on the support vectors that are selected as the only effective elements in the training set. Thus, the goal of the Support Vector Machiness learning is to find the optimal

separating hyperplane that has the maximal margin to both sides (Mertsalov and McCreary, 2009).

For example, to classify documents into positive and negative sets, the SVM algorithm learns to distinguish between the two categories based on a training set of documents that contains labeled examples from both categories. SVM manipulates documents to represent them as points in a high dimensional space and then finds a hyperplane that optimally separates the two categories (Mertsalov and McCreary, 2009).

2.6.2.3 Arabic Text Classification

El-Kourdi et al. (2004), utilised the Naïve Bayes algorithm to conduct an automated classification of news documents with 0.92 accuracy. Alsaleem (2011) investigated Naïve Bayes and SVM on different Arabic data sets, finding that the SVM algorithm outperformed Naïve Bayes with F_1 score of 0.77 and 0.74 respectively. Baraka et al. (2014) used SVM for the problem of author identification for Arabic text. They performed several experiments on Arabic text documents taken from two domains: politics and literature. The accuracy was almost 1.00 for text document classification for these domains. Mohammad et al. (2016) used the three text classification algorithms SVM, NB, and Multilayer Perceptron Neural Network (MLP-NN) algorithms on a large Arabic news data set. The results showed that SVM yields the best results with average F_1 score of 0.77. Altamimi and Teahan (2017) applied MNB, K-Nearest Neighbours, SVM and PPMD, a specific variant of PPM, to gender and authorship categorisation for Arabic text taken from Twitter. The experiments showed that PPMD had significantly better 0.90 and 0.96 accuracy than all the other algorithms for gender and

authorship respectively.

2.6.3 Dyslexia Classification

Dyslexia can be identified using a variety of techniques. Perera et al. (2016) suggest that these techniques can be classified as follows: the examination of ‘behavioural’ symptoms and primary features, including reading and spelling; the use of brain imaging methods to visualise distinctive brain behaviours and finally, the analysis of the nature of the individual’s eye movement patterns.

Several studies (Frid and Breznitz, 2012; Karim et al., 2013; Zainuddin et al., 2016) on identification dyslexia describe using the Electroencephalogram, commonly known as EEG, as a technique that can be used to monitor and detect brain functions. The electrical activity of the brain for various stimuli can be identified via the electrodes placed on the scalp (Nunez and Srinivasan, 2006). On the other hand, some researchers have employed natural language processing and machine learning techniques to identify dyslexia. In their study, Kohli and Prasad (2010) adopted an approach for the identification of dyslexia that relied on Artificial Neural Networks (ANNs). Rello and Ballesteros (2015) found that eye-tracking technology measures, based on a model founded on an SVM binary classifier, could be used to predict whether a reader had dyslexia. Furthermore, Tamboer et al. (2016) examined whether or not it is possible to differentiate young adults with and without dyslexia. The study analysed the neuroanatomical networks associated with dyslexia, a process which relies on the utilisation of a whole-brain class employing SVM and cross-validation. Al-Barhamtoshy and Motaweh (2017) diagnosed dyslexia using computing analysis techniques based on

metrics related to individual's results on the Gibson test. The Gibson test records brain skills used to measure a number of factors related to dyslexia such as writing. The method classified the test dataset they used into non-dyslexic, dyslexic, or other disorders (inattention, hyperactivity, or other) using three classifiers – K-means, ANN, and Fuzzy. Khan et al. (2018) used the K-Nearest Neighbours classifier algorithm to distinguished two types 'no dyslexia' and 'seems to be dyslexic' – in spelling and reading.

There is a general lack of research specifically focussed on classifying Arabic dyslexic texts. As far as the researcher knows, no previous study used PPM to classify dyslexic texts in Arabic.

2.6.4 Spelling Correction

Issues relating to spelling error correction have been investigated by many researchers for several decades and it remains a topic of interest to natural language processing researchers. The first study was carried out by Damerau in 1964. He developed a technique for detection and correction of spelling errors based on omission, addition, substitution and transposition. Kernighan et al. (1990) and Church and Gale (1991) used the noisy channel model technique (described below) for the purpose of spellchecking. Kukich (1992) divided spelling correction into three types: error detection; isolated word correction; and context-sensitive correction. In 2000, Brill and Moore described an error model for noisy channel spelling correction based on string-to-string edits.

There are different approaches that can be used to solve the problem of spelling error detection and correction. These can be categorised as lexicon based, rule based, statistical based and combination of these approaches.

The next sections specifically discuss the noisy channel model and n-gram models in more detail which are statistical based.

2.6.4.1 The Noisy Channel Model

The noisy channel model is an approach used for many of the most commonly used natural language processing tasks, such as OCR, spelling correction, POS tagging and machine translation (Kernighan et al., 1990). The noisy channel model is based on a theoretical model for the input of a sequence of text, which is processed through a communications channel, where noisy text is produced. For example, ‘noisy text’ can be used to denote text with error words.

Teahan (1998) described how the noisy channel technique is applied to the spellchecking process. Formally, a sequence of text S is processed through a communications channel. As a result, a noisy text O is produced. The fundamental issue in this regard is deducing the correct form of the input text by using the output text as a base for this deduction. This can be facilitated by making hypothesised versions of the input text S and then choosing which version \hat{S} seems most likely based on the given output text O :

$$\hat{S} = \arg \max_S p(S | O). \quad (2.6)$$

The intuition is to apply Bayes’ theorem to transform equation 2.6 into a set of other probabilities (Jurafsky and Martin, 2018) as shown in equation 2.7.

$$\hat{S} = \arg \max_S \frac{p(S)p(O | S)}{p(O)}. \quad (2.7)$$

Equation 2.7 can be simplified by dropping the denominator $p(O)$ as shown in equation below 2.8 (Jurafsky and Martin, 2018). This is because $p(O)$ does not change for each version of the input text since the processing always requires trying to find the most likely input for the same output text (observed error) O , which must have the same probability $p(O)$ (Jurafsky and Martin, 2018).

$$\hat{S} = \arg \max_S p(S)p(O | S). \quad (2.8)$$

According to this equation, the way to determine the output with the highest probability is through the parameter $p(S)$, denoting the prior probability that the pertinent sequence will appear together with the observation (or channel) probability $p(O|S)$. The prior probability $p(S)$ is usually not available, so a model is used instead.

2.6.4.2 n-gram Language Models

n-grams are sequences extracted from a text that may be in the form of words or characters (Majumder et al., 2002), where n can be any digit starting from 1; therefore, commonly used n-grams include a unigram ($n=1$), bigrams ($n=2$) and trigrams ($n=3$) (Zamora et al., 1981).

The word n-gram language model calculates the probability of a word based on the previous $n-1$ immediately preceding words in the text (Jelinek, 1990).

Let $p(S)$ be the probability of a sequence S of n words $w_1, w_2, w_3, \dots, w_n$ given by the equation as follows:

$$\begin{aligned} p(S) &= p(w_1)p(w_2 \mid w_1)p(w_3 \mid w_1, w_2)\dots p(w_n \mid w_1, \dots, w_{n-1}). \\ &= \prod_{i=1}^n p(w_i \mid w_1, w_2, \dots, w_{i-1}). \end{aligned} \quad (2.9)$$

Here, w_i is the word being predicted and w_1, w_2, \dots, w_{i-1} is the history or conditional context. Clearly, using a full history to build the language model would be computationally expensive. This problem can be solved using the Markov assumption, where the conditioning context is considered equivalent to the preceding $w-1$ words. For example, a trigram model uses the previous two words in order to determine probability:

$$p(S) \approx \prod_{i=1}^n p(w_i \mid w_{i-2}, w_{i-1}). \quad (2.10)$$

Models such as the bigram and trigram models that use the Markov assumption to predict the next word are referred to as Markov models. Generally, an n -gram model is known as an order $n-1$ Markov model (Teahan, 1998).

However, many n -grams may not appear in the training data at all. As a consequence, a zero probability is assigned to these n -grams. Smoothing techniques prevent the model from having a zero probability, for example by using a ‘backing-off’ or ‘escaping’ technique as occurs with PPM (Teahan, 1998). PPM is explained later in Section 2.7.1.

2.6.4.3 Arabic Spelling Correction

The study conducted by Mars (2016) developed a system for automatic Arabic text correction based on a sequential combination of approaches including lexicon based, rule based and statistical based. The F_1 score obtained through the study was 0.67, 0.73 precision and 0.65 recall. Likewise, AlShenaifi et al. (2015) used the rule-based, statistical-based and lexicon-based approaches in a cascade fashion, with an F_1 score of 0.57, recall 0.51, and precision 0.66.

Another study by Mubarak and Darwish (2014) employed an approach based on two correction models and two punctuation recovery models: a character-level model and a case-specific model, a simple statistical model and a conditional random fields model, respectively. The best result was by using a cascaded approach that involves a character-level model, then case-based correction, resulting in an F_1 score of 0.63, precision of 0.71 and recall of 0.56. Alkanhal et al. (2012) used the Damerau–Levenshtein edit distance to generate alternatives for each error word. The selection-based method was then used on the maximum marginal probability via an A* lattice search and n-gram probability estimation to select the most applicable word. For error word detection, the experimental result showed an F_1 score of 0.98, precision of 0.99 and a recall of 0.97. In terms of correction of error words, the system achieved an F_1 score of 0.92, recall of 0.88 and precision of 0.96.

Conversely, Zaghouani et al. (2015) used regular expression patterns to detect errors by using the Arabic verb forms and affixes and built a rule-based correction method that added linguistic rules using existing lexicons and regular expressions to correct native and non-native text. The system achieved an F_1 score of 0.67, precision 0.84 and recall 0.56 for native speakers; and

an F_1 score of 0.32, precision 0.59 and recall 0.22 for non-native speakers. Similarly, Nawar and Ragheb performed two studies in 2014 and 2015. The first study developed a rule-based probabilistic system, which achieved an F_1 score of 0.65 on the data (Nawar and Ragheb, 2014). In 2015, Nawar and Ragheb made improvements to a previous statistical rule-based system in which word patterns were used to improve error correction. They also used a statistical system using the syntactic error correction rules; the system achieved an F_1 score of 0.72 on a dataset containing Aljazeera articles by native Arabic speakers and an F_1 score of 0.35 on the non-native speakers' data. Mubarak et al. (2015) employed a case-specific correction approach that addressed particular errors such as substitution and word splits and some errors that are specific to non-native speakers such as gender-number agreement. The best result on non-native speakers' data gave an F_1 score of 0.27, precision of 0.46 and a recall 0.19.

Some studies have adopted the noisy channel model approach. Shaalan et al. (2012) detected errors by building a character-based trigram language model in order to classify words as valid and invalid. For correction, they used finite-state automata to propose candidate corrections within a specified edit distance measured by Levenshtein distance from the error word. After choosing candidate corrections, they used the noisy channel model and knowledge-based rules to assign scores to the candidate corrections and choose the best correction independent of the context. Additionally, Noaman et al. (2016) used pairs of spelling errors and a corrected form extracted from the Qatar Arabic Language Bank (QALP) to build an error confusion matrix, then used this confusion matrix with the noisy channel model to generate a candidates' list and select a suitable candidate for the erroneous word. The overall system accuracy that was obtained was 0.85. On the

other hand, a study by Attia et al. (2012) attempted to improve three main components: the dictionary, the error model and language model. The way they improved the error model was by analysing error types and creating an edit distance-based re-ranker that analysed the level of noise in different sources to improve the language model. By improving the three main components, they achieved an accuracy rate of 0.83.

2.6.4.4 Dyslexia Spelling Correction

There are few studies that deal with dyslexic spelling correction. Pedler (2007) developed a program to detect and correct dyslexic real-word spelling errors in English. The method identifies sets (often pairs) of words that are likely to be confused, such as *loose* and *lose*, and then, when encountering one of the words (*loose*) in the text being checked, determines whether the other one (*lose*) would be more appropriate in the context. The first stage of the approach adopted by Pedler (2007) considered words that differed in their parts-of-speech. Decisions for words that have the same parts-of-speech were left for the second stage, which used semantic associations derived from WordNet. The program achieved precision with 0.80 and 0.44 recall. Rello et al. (2015) used a probabilistic language model, a statistical dependency parser and Google n-grams to detect and correct real-word errors in Spanish in a system called Real Check. The system achieved a F_1 score of 0.57 for the detection and F_1 score of 0.33 of the correction. For the Arabic language, Alamri (2013) used the PPM model to correct the spelling errors of writers with dyslexia in Arabic texts. The accuracy was 0.67 for the correction single-character errors.

As mentioned earlier in Section 2.6.4.3, there have been attempts made to

detect and correct Arabic text in general. However, there is a general lack of research related specifically to the problem of correcting dyslexic texts in Arabic.

The next section will discuss text compression as that is related to the solutions that have been adopted in this thesis.

2.7 Text Compression

Techniques used in data compression are split into types: lossless and lossy. Lossy compression techniques result in a loss of some information, which means that the data cannot usually be reconstructed or recovered in its entirety. Lossless compression, in contrast, avoids the loss of any information in the compression process, so the original data can be recovered in its entirety. This type of compression is mainly used in applications that require the recovered and original data to be identical. Text compression is an important example (Sayood, 2017).

The implementation of lossless text compression is usually achieved through the use of either statistical- or dictionary-based compression (Nelson and Gailly, 1996). In dictionary-based text compression, groups of consecutive characters or symbols in the text are replaced by a code (Bell et al., 1989) such as Lempel-Ziv. Statistical-based text compression as such PPM relies on the probabilities of each symbol occurring (a symbol may be a character or a word); more frequent and therefore more probable symbols are encoded using fewer bits (Nelson and Gailly, 1996). The use of statistical methods (compression-based language models) has been fine-tuned for over 30 years and has been proven fruitful, as evidenced by the results of several different

natural language processing applications (Teahan, 2018). This thesis makes use of lossless text compression, more specifically the Prediction by Partial Matching (PPM) method of compression. Therefore, the next subsection will discuss PPM in more detail.

2.7.1 Prediction by Partial Matching

PPM is an adaptive, statistical method of compression designed by Cleary and Witten in 1984. A statistical model sequentially processes the symbols (typically characters) that are currently available in the input data (Teahan, 2000).

PPM uses the past few characters in an input stream to predict the next one (Teahan et al., 1998). The encoder uses the input data to predict probability distributions for new symbols. An arithmetic encoder is subsequently used to encode new symbols with a predicted probability. PPM comprises two different processes: modelling and coding. The model generates a probability distribution of the symbols that may occur next based on the symbols seen before in the text, while the coder is used to encode the symbol that actually occurred using this probability distribution (Teahan, 1998). Fewer bits will be required by the arithmetic encoder to encode the symbol if this probability is higher, and the compression performance will also be more efficient.

The conditioning contexts in PPM are finite sequences of symbols that precede the current symbol being predicted. PPM uses a Markov-based approach, the purpose of which is to utilise the immediately preceding characters of an input stream for the prediction of what will come next. The PPM model's order is the maximum context length used to predict the next

symbol.

PPM has been applied to various applications in natural language processing. As reported by Teahan (1998), a fixed order context of five is usually the most effective for compression of English text. However, other researchers have found different orders depending on the application giving the most effective result. For example, Teahan et al. (2000) used PPM for Chinese segmentation and found that an order 3 model yielded the best result. For Thai segmentation, Sornil and Chaiwanarom (2004) found that an order 4 model gives the best results. Frank et al. (2000) found order 2 the most effective order for English topic categorisation.

2.7.1.1 Variants of Prediction by Partial Matching

There are several variations of PPM, such as PPMA and PPMB proposed by Cleary and Witten (1984), PPMC by Moffat (1990) and PPMD by Howard (1993), depending on the methods proposed for calculating symbol probabilities, each differs by the escape method used. For example, PPMC uses escape method C, and PPMD uses escape method D. Also, the maximum order of the context models may be included when the variant is described; for example, PPMD2 refers to a fixed-order 2 PPM model using escape method D. Previous experiments showed that PPMD, in most cases, performs better than the other variants. Thus, the adoption of PPMD is explored in this thesis. All the PPM experiments reported later were performed using the Tawa Toolkit (Teahan, 2018). The toolkit is based on an earlier tool designed by Cleary and Teahan (1997) for modelling text using text compression models. The Tawa toolkit provides a method for calculating compression codelengths, and for classifying and transforming text (Teahan,

2018). An example illustrating how the PPMD variant works is provided below.

The equation below can be applied to calculate the probability p of the symbol s for PPMD:

$$p(s) = \frac{2c(s) - 1}{2T}. \quad (2.11)$$

where T is the total number of times that the current context has happened and $c(s)$ is the number of times the current context was followed by the symbol s (Howard, 1993).

A problem occurs (called the ‘zero frequency problem’) when the current context cannot predict the upcoming symbol. In this case, PPM ‘escapes’ or ‘backs off’ to a lower order model where the symbol has occurred in the past. If the symbol has never occurred before, then PPM will ultimately escape to what is called an order -1 context where all symbols are equiprobable. The escape probability e for PPMD is estimated as follows:

$$e = \frac{t}{2T}. \quad (2.12)$$

where t represents the total number of times that a unique character has appeared after the current context.

As an example, the PPMD model after the string ‘*dyslexicornotdyslexic*’ has been processed is shown in Table 2.6 below. For demonstrative purposes, a maximum model order of 2 has been adopted for this example, where *context* \rightarrow *character* represents the prediction, c represents the count, p represents the probability and A represents the size of the alphabet.

Table 2.4: The PPMD model after the string ‘*dyslexicornotdyslexic*’ has been processed.

Order 2			Order 1			Order 0			Order -1		
Prediction	c	p	Prediction	c	p	Prediction	c	p	Prediction	c	p
dy → s	2	$\frac{3}{4}$	d → y	2	$\frac{3}{4}$	→ d	2	$\frac{3}{42}$	→ A	1	$\frac{1}{ A }$
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ y	2	$\frac{3}{42}$			
ys → l	2	$\frac{3}{4}$	y → s	2	$\frac{3}{4}$	→ s	2	$\frac{3}{42}$			
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ l	2	$\frac{3}{42}$			
sl → e	2	$\frac{3}{4}$	s → l	2	$\frac{3}{4}$	→ e	2	$\frac{3}{42}$			
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ x	2	$\frac{3}{42}$			
le → x	2	$\frac{3}{4}$	l → e	2	$\frac{3}{4}$	→ i	2	$\frac{3}{42}$			
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ c	2	$\frac{3}{42}$			
ex → i	2	$\frac{3}{4}$	e → x	2	$\frac{3}{4}$	→ o	2	$\frac{3}{42}$			
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ r	1	$\frac{1}{42}$			
xi → c	2	$\frac{3}{4}$	x → i	2	$\frac{3}{4}$	→ n	1	$\frac{1}{42}$			
→ <i>Escape</i>	1	$\frac{1}{4}$	→ <i>Escape</i>	1	$\frac{1}{4}$	→ t	1	$\frac{1}{42}$			
ic → o	2	$\frac{1}{2}$	i → c	2	$\frac{3}{4}$	→ <i>Escape</i>	12	$\frac{12}{42}$			
→ <i>Escape</i>	1	$\frac{1}{2}$	→ <i>Escape</i>	1	$\frac{1}{4}$						
co → r	2	$\frac{1}{2}$	c → o	1	$\frac{1}{2}$						
→ <i>Escape</i>	1	$\frac{1}{2}$	→ <i>Escape</i>	1	$\frac{1}{2}$						
or → n	2	$\frac{1}{2}$	o → r	1	$\frac{1}{4}$						
→ <i>Escape</i>	1	$\frac{1}{2}$	→ t	1	$\frac{1}{4}$						
rn → o	1	$\frac{1}{2}$	→ <i>Escape</i>	2	$\frac{2}{4}$						
→ <i>Escape</i>	1	$\frac{1}{2}$	r → n	1	$\frac{1}{2}$						
no → t	2	$\frac{1}{2}$	→ <i>Escape</i>	1	$\frac{1}{2}$						
→ <i>Escape</i>	1	$\frac{1}{2}$	n → o	1	$\frac{1}{2}$						
ot → d	2	$\frac{1}{2}$	→ <i>Escape</i>	1	$\frac{1}{2}$						
→ <i>Escape</i>	1	$\frac{1}{2}$	t → d	1	$\frac{1}{2}$						
td → y	2	$\frac{1}{2}$	→ <i>Escape</i>	1	$\frac{1}{2}$						
→ <i>Escape</i>	1	$\frac{1}{2}$									

If the next character in the string to be encoded is *o*, we must make the prediction *ic*→*o* using the order 2 context. Since the character *o* has been seen once before in the context *ic*, then a probability of $\frac{1}{2}$ will be assigned by using equation 2.11 as $c = 1$. Correspondingly, 1 bit will be required by the encoder to encode the character.

However, if the subsequent character has not previously been seen in the order 2 context (i.e. presuming the next letter would be *n* instead of *o*,

say), it will be necessary to conduct an escape procedure or back off to a lower order. In this case, the escape probability will be $\frac{1}{2}$ (calculated by equation 2.12), and a lower order of 1 will then be applied by the model. When this happens, the character n is also found not to be present after c . As a result, the model will need to encode a further escape (whose probability will also be estimated as $\frac{1}{2}$), and there will be a reduction in the current context to order 0. In this order, the probability that will be applied to encode letter n will be $\frac{1}{42}$. The total cost of predicting this letter is $\frac{1}{2} \times \frac{1}{2} \times \frac{1}{42} = \frac{1}{168}$, which costs around 7.39 bits to encode it ($-\log_2 \frac{1}{168} \approx 7.39$).

If on the other hand the next character has never been seen before and is appearing for the first time (such as letter u), beginning at the maximum order of 2 and escaping down to order -1, the model will apply the following probabilities encoding an escape three times: $\frac{1}{2} \times \frac{1}{2} \times \frac{12}{42} \times \frac{1}{256}$, where 256 is used for the size of the alphabet for the English language encoded using 8-bit ASCII, thus requiring approximately 11.80 bits to conduct the encoding process.

Improvements in prediction are possible by two mechanisms: full exclusions and Update Exclusions (UE). Full exclusions result in higher order symbols being excluded when an escape has occurred (Teahan and Cleary, 1997), while Update Exclusions (UE) (Moffat, 1990) only update the counts for the higher orders until an order is reached where the symbol has already been encountered (Teahan, 1998). This mechanism typically improves the compression rate by up to 2% as stated by Bell et al. (1990). On the other hand, when PPM is applied Without Update Exclusions (WUE), all the counts for all orders of the model are updated. The counts are incremented even if they are already predicted by a higher order context.

PPM is an adaptive compression method which means there is an issue at the beginning where the models are empty with not enough data to effectively compress. A suitable solution is to prime the models by using training texts (corpora) that are representative of the text being compressed. This thesis use static models that means that when processing the testing text, no further updates occur, as the models have been primed using the training texts.

2.7.1.2 Compression Codelengths using PPM-Based Models

The codelength is the number of bits required to encode the text using the PPM compression model. PPM codelength is the length of the compressed text, in bits, when it has been compressed using the PPM language model. The smaller the codelength value, the more closely the text resembles the text used to train the language model. It can be used to calculate the codelength ratio of the text by dividing the size of the text begin compressed.

The $C|C^2$ model which is labeled in Table 2.5, represents an order 2 PPM character model, where the predictions are based on the stream of character symbols. p' is the probabilities estimated by the order two PPM model. So, the probability of S (where S is the sequence of length n characters c_i) is given by:

$$p(S) = \prod_{i=1}^n p'(c_i | c_{i-2} c_{i-1}) \quad (2.13)$$

The compression codelength can be calculated according to the following

Table 2.5: Models for predicting character streams (Teahan, 1998).

The symbol \hookrightarrow in the table represents an escape

C C² Model
$p(c_i c_{i-2}c_{i-1})$
$\hookrightarrow p(c_i c_{i-1})$
$\hookrightarrow p(c_i)$
$\hookrightarrow p_{eq}(c_i)$

equation:

$$H(S) = -\log_2 p(S) = -\log_2 \prod_{i=1}^n p'(c_i|c_{i-2}c_{i-1}) \quad (2.14)$$

where $H(S)$ is the number of bits required to encode the text.

2.7.1.3 Text Classification Using PPM

The classification method is based on the idea that a character based approach for compressing the texts can be used to help classify the texts. Essentially, the classifier can adopt the style or type of text associated with the compression model that compresses the text best where each model is trained or primed using representative training text for each style or type.

PPM codelength has been explained in detail in Section 2.7.1.2. Moreover, it can be used to estimate the codelength ratio of the text using a PPM model by the following equation:

$$H_M(T) = -\frac{1}{n} \sum_{i=1}^n \log_2 p'_M(x_i|x_{i-m}, \dots, x_{i-1}). \quad (2.15)$$

where H is the codelength ratio as measured by the PPM compression ratio given model M of order m for the probability distribution for the symbols x_i over the text sequence $T = x_1, x_2, \dots, x_n$ of length n . Each symbol will be

predicted based on the prior context x_{i-m}, \dots, x_{i-1} of length m using the Markov-based approach.

Text classification using PPM is processed by training (priming) N different models M_1, M_2, \dots, M_N where N is indicated by the number of classes and the training text used to prime each model is representative of the class being modelled. The following equation describes how to guess the correct class of the text T for each class i :

$$\hat{\theta}(T) = \arg \min_i H_{M_i}(T). \quad (2.16)$$

Basically, a PPM model is built for each class, and the testing text is compressed using each model with the class being chosen from the model that compresses the text best, and the best model is judged to be the one with the minimum codelength ratio (Teahan, 1998).

2.7.1.4 PPM Compression Method for Arabic

To explain the process of the PPM method for Arabic, Table 2.6 illustrates the state of the PPMD model of order 2 after the string “المعلم” has been processed. Again for purposes for this example, the maximum context order is 2. If the next character is estimated successfully by the modelling context, the probability p will be used to encode it, while c denotes the occurrence counts.

Concerning the example, if the input string “المعلم” is followed by the character ‘ع’, the probability of the prediction ‘ع’ → ‘م’ in order 2 is $\frac{1}{2}$ would be used to encode it, requiring only one bit as a result ($-\log_2 \frac{1}{2} = 1$).

Assume instead that ‘ل’ follows the string ‘المعلم’. As the order 2 model does

Table 2.6: The PPMD model after the string ‘المعلم’ has been processed.

Order 2			Order 1			Order 0			Order -1		
Prediction	c	p	Prediction	c	p	Prediction	c	p	Prediction	c	p
ا → م	1	$\frac{1}{2}$	ا → ج	1	$\frac{1}{2}$	→ ا	1	$\frac{1}{12}$	→ A	1	$\frac{1}{ A }$
→ Escape	1	$\frac{1}{2}$	→ Escape	1	$\frac{1}{2}$	→ ج	2	$\frac{3}{12}$			
ل → ع	1	$\frac{1}{2}$	ج → م	2	$\frac{3}{4}$	→ م	2	$\frac{3}{12}$			
→ Escape	1	$\frac{1}{2}$	→ Escape	1	$\frac{1}{4}$	→ ع	1	$\frac{1}{12}$			
م → ج	1	$\frac{1}{2}$	م → ع	1	$\frac{1}{2}$	→ Escape	4	$\frac{4}{12}$			
→ Escape	1	$\frac{1}{2}$	→ Escape	1	$\frac{1}{2}$						
عل → م	1	$\frac{1}{2}$	ع → ج	1	$\frac{1}{2}$						
→ Escape	1	$\frac{1}{2}$	→ Escape	1	$\frac{1}{2}$						

not predict this character, the escape probability of $\frac{1}{2}$ will be encoded for this order, and the encoder will escape from the order 2 model down to the order 1 model. As the order 1 model does not predict this character, the escape probability of $\frac{1}{2}$ will be encoded for this order, and the encoder will escape from the order 1 model down to the order 0 model. In this context, \rightarrow ‘ا’ predicts the character ‘ا’, with a probability of $\frac{1}{12}$. Thus, the total probability needed to encode the ‘ا’ character is $\frac{1}{2} \times \frac{1}{2} \times \frac{1}{12}$, which requiring $(-\log_2 \frac{1}{48} = 5.58 \text{ bits})$.

In order to deal more effectively with Arabic texts, in which each character needs two or more bytes to be represented when using UTF-8 encoding, a method called Character Substitution of Arabic for PPM (CSA-PPM) was presented by Alhawiti (2014). There are two important operations in this method, which are pre-processing and post-processing, used in conjunction with the PPM method. Each two-byte Arabic character is substituted with an equivalent number of the UTF-8 encoding scheme in the pre-processing operation and, as a result, one output file is generated while the post-processing is performed by replacing the numbers with the original equivalent characters. The use of this method has not only shown a considerable improvement in Arabic text compression but also for other texts that

use Arabic script, such as Persian and Kurdish.

The Buckwalter Arabic transliteration is another method commonly used. It is defined as an ASCII transliteration that is used to represent Arabic texts for computers for standard Arabic encoding schemes. In this transliteration scheme, Arabic orthographic letters are substituted one to one. For instance, the Arabic word “المعلم” is represented by the letters [B: “AlmElm”] (Habash et al., 2007). Al-kazaz (2018) conducted an experiment to check the performance of the “CSA-PPM” and the Buckwalter transliteration methods. The results indicate that using either of these two methods for the Arabic language shows a significant improvement in the codelength compression ratio. Moreover, the compression ratio produced by both the Buckwalter transliteration and CSA-PPM methods are similar.

2.8 Conclusion

This chapter has discussed the issues related to the aim and objectives of this research. This included in-depth research on dyslexia and Arabic language in addition to spelling errors in general and spelling errors by people with dyslexia. The concepts of corpora was also discussed in detail.

This chapter has also reviewed concepts of two natural language processing applications, which were text classification and spelling correction. These were discussed as they are the two fundamental objectives of this study. Furthermore, data compression was discussed and the differences between text compression types were discussed (lossless and lossy). Moreover, the PPM text compression algorithm was described in detail, and how the codelength metric is calculated.

Chapter 3

Bangor Dyslexic Arabic Corpus

3.1 Introduction

Corpora have made a significant contribution to the understanding of language and teaching methods (McEnery and Xiao, 2010), and can be employed by instructors not only when considering what to teach, but also when evaluating what learners may learn directly (Gavioli and Aston, 2001). A need currently exists in the Arabic context for a dyslexia corpus to serve as a basis for studies of Arabic dyslexia, in terms of both dyslexia in the Arabic language in general, and more specifically in the context of children with dyslexia.

The objective of this chapter is to enlarge the Bangor Dyslexia Arabic Corpus (BDAC) (Alamri, 2013), and the chapter therefore reviews the process of creating a dyslexia corpus for Arabic. As the fieldwork was conducted

in Saudi Arabia, it is required to provide an overview of how dyslexia is identified in Saudi Arabian schools Section 3.2. Section 3.3 discusses the dyslexic texts collected for this study, and explains how the material was gathered. Since the majority of the material was hand-written, it was necessary to transcribe it into an electronic form, and this process is described in Section 3.4. Meanwhile, Section 3.5 describes the type and size of the BDAC corpus, and the textual information, in terms of its source and categorisation. The final two sections present the participant information, including the age and gender of the dyslexic participants, and finally an analysis of the corpus data, by examining the frequency of the words and characters.

3.2 Identifying Dyslexia in Saudi Arabian Schools

According to the special education teachers' policy guidelines (Ministry of Education of Saudi Arabia, 2015), students with learning difficulties are usually identified at home by their parents, or at school by their teacher, when the student exhibits certain patterns of behavioural or psychological characteristics.

There are two ways in which students with learning difficulties are identified in Saudi Arabia. First, when the child's general class teacher, parent, or other professional, such as a social worker, notices that the student appears to lack the ability to perform certain required tasks, and depends on the help of others to conduct these tasks, which often engenders failure in their academic achievement, and reluctance to attend school, together with a withdrawal from classroom participation. This behaviour signals that the student may require an intervention, and a follow-up with a specialist. The second way in which students with learning difficulties are identified in Saudi

Arabia involves special education teachers, who assesses the academic results of the students who exhibit a poor performance compared with their peers. This underperformance might be an indicator that the student has a learning difficulties.

The student is then assessed using formal measures, such as an IQ test, and informal measures, such as curriculum-based tests or dyslexia checklist. These tests determine the nature of the learning difficulties, and identify whether the student is dyslexic, or has dyscalculia, or other difficulties.

Once a student is diagnosed with a learning difficulties, they become eligible for assistance from special education teachers within the education service. They continue to be educated alongside their typically developing peers in general education classrooms, but also receive extra help from special education teachers in a resource room. A resource room is defined as a room in an ordinary school that students with learning difficulties attend for a period of not more than half of the school day, for the purpose of receiving special education services from a special education teacher (Ministry of Education of Saudi Arabia, 2015).

3.3 Data Collection Procedure

As explained in Section 3.2, the students with dyslexia in Saudi Arabia are identified in school. Therefore it was deemed appropriate to collect the data from schools for this study.

The researcher visited schools with a resource room in Riyadh, since it is the capital city of Saudi Arabia with a significant number of resource rooms. In order to commence the data collection, it was necessary to obtain the

relevant permission from the Ministry of Education of Saudi Arabia (see Appendix 1). In addition, permission was obtained from the relevant teachers and the parents of the student participants, all of whom were required to sign a consent form (see Appendix 2) granting permission to use their child's texts for the purposes of the study. The form confirmed that the participants' information would remain anonymous except age and gender.

The researcher collected 25,248 words of the corpus content during this stage. However, it was more challenging and complicated to obtain this number of words from the schools than expected, since some of the teachers or head teachers were more cooperative than others, and transport to the schools, together with locating the schools, also proved to be challenging. Due to these difficulties, the researcher decided to employ additional collection methods to enlarge the corpus. The second method involved collecting data from the parents of a student with dyslexia who received services from the resource room. This collection method expanded the corpus to 26,541 words. An additional third method distributed a form (see Appendix 3) to males' teachers of dyslexic students to asked their students to complete. The texts subsequently received increased the corpus to 27,136 words, plus those taken from the researcher's previous work, which consisted of 1,067 words, resulting in a total size for the BDAC corpus of 28,203 words.

3.4 Transcribing the Handwritten Data

The texts were handwritten, and it was therefore necessary to transcribe them into an electronic format, in order to conduct the analysis of the dyslexics text. All the text was manually entered into a spreadsheet via a Google form by the researcher and one volunteer. Figure 3.1 presents a

screenshot of the text transcription form, showing that the data was entered into the spreadsheet according to the following fields: age, gender, text source, category, raw text, and correct text. The age field was numerical, while the other fields included textual elements in Arabic and English. The raw text is the text that was written by the people with dyslexia while the correct text is the intended text. Some teachers corrected the text that dyslexic students wrote (see Figure 3.2). However, any text that did not include the corrected text required further work to either locate the correct text, or to choose the word in accordance with the written text as much as possible.

According to Alfaifi (2015), there is no standard practice for transcribing Arabic from a handwritten format into a computerised form, and he therefore developed a series of standards (see Appendix 4) for achieving a high level of consistency during transcription. This study employed these standards in order to ensure a greater consistency and improved compatibility with the dyslexia errors, with an edited version of some points as follows:

1. Any struck-out text should be excluded.
2. If there is a correction above a non-struck-out word, the ‘first’ form should be transcribed or if a word has been written more than once within the same document in an attempt to achieve the correct form, the first version should be transcribed.
3. When there is a doubtful form of a character, the form closest to the correct form should be transcribed.
4. If there is an overlap between handwritten characters, which cannot be transcribed, the closest possible form should be selected.

Text Transcription

* Required

Age - العمر *

Your answer

Gender *

- ☐ Male - ذكر
- ☐ Female - انثى

Text Source - النص من *

- ☐ HW - الواجب
- ☐ Parent - الوالدين
- ☐ Form - نموذج

Category - الصنف *

- ☐ Word - كلمة
- ☐ Sentence - جملة
- ☐ Paragraph - فقرة

Raw Text - النص المكتوب *

Your answer

Correct Text - النص الصحيح *

Your answer

SUBMIT

Never submit passwords through Google Forms.

Figure 3.1: Screenshot of the text transcription form.

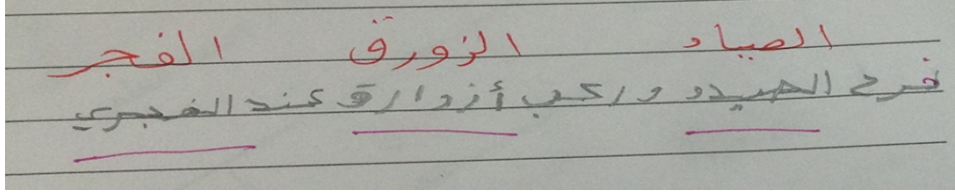
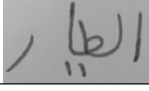


Figure 3.2: Example of a handwritten text written by a girl with dyslexia with the intended text transcribed by a teacher.

5. If a writer forgot to add a character's dot(s), whether above or below, it should be transcribed as written by the learner, unless this is not possible. For example, if there is no equivalent character on the computer.
6. Any identifying information (e.g. learner's name, contacts, postal address, etc.), should be excluded. Other non-personal information can be left such as class, city, country, religion, culture, etc.
7. Any shape, illustration, or ornamentation drawn by the learner on the sheet is excluded.
8. Any text formatting should be excluded, such as underlined words or sentences.
9. Unknown words or phrases should be removed.
10. If there is more than one space between characters within one word, transcribe as one space.
11. If a writer uses an incorrect letter form, transcribe if possible; otherwise, write it as the correct letter shape. For example, the letter . written at the beginning as if it is in middle letter of the word.
12. Words that cannot be transcribed, either because of difficulty reading the handwriting, or because it was not possible to transcribe them,

should be removed. Table 3.1 below presents some examples.

Table 3.1: Examples of words for which it was not possible to transcribe.

Error word	Description
	كتابة حرف الياء كما لو أنه في نهاية الكلمة [Writes letter ي as if it is in end of the word].
	كتابة حرف الباء كما لو أنه ياء [Writes letter ب as letter form ي].
	كتابة نقاط حرف الياء بدون وجود حرف الياء [Writes dots of letter ي without written letter ي].
	خطأ في كتابة حرف القاف [Wrong written form of ق letter].
	كتابة حرف النون كما لو أنه ياء [Writes letter ن as letter ي].
	خطأ في كتابة حرف السين [Wrong written form of س letter].
	كتابة بصورة معكوسة [Mirror writing].
	صعوبة في معرفة ما المقصود من هذه الكلمة [Hard to guess the intended word].

3.5 The Bangor Dyslexia Arabic Corpus

As discussed in the literature review (Section 2.5.1), different types of corpora exist. However, the BDAC corpus created in this study is a specialised

corpus, as its content was collected only from people with dyslexia.

As also mentioned in the literature review (Section 2.5.2), there are dyslexia corpora in non-Arabic language such as English, 12,000 words (Pedler, 2007) and Spanish, 1,057 words (Rello et al., 2012). These studies confirmed that the corpus of around 1,000 errors can yield useful results. This view agrees with Biber (1993), who argued that a sample of 1,000 words may prove sufficient for studying features, such as the amount of present and past tense verbs in English. Moreover, Shalom (1997) conducted a study of a 20,000 word corpus of personal adverts, which tend to be brief, and also reasonably repetitive, and was able to identify the patterns of language employed across the different adverts despite this relatively small size. A more recent study conducted by Xiao (2010) revealed that a small corpus can include sufficient examples to highlight a linguistic phenomenon.

Therefore, this justifies the size for the BDAC corpus as a useful size for the study, with the corpus ultimately consisting of 28,203 words.

3.5.1 Text Resources of the BDAC Corpus

The data for this study was gathered from the following sources: *Homework* for text from the dyslexic notebook; *Parent* for text provided by the parents of children with dyslexia, and *Form* for text gathered using a form designed by the researcher, which included questions answered by participants with dyslexia. All the texts collected for this study were written in modern standard Arabic.

The content of the BDAC corpus consisted of 26 responses to the form, 11 samples provided by parents, and 867 samples taken from the students'

homework. The example of the Homework resource is illustrated in Figure 3.3. More examples are provided in Appendix 5.

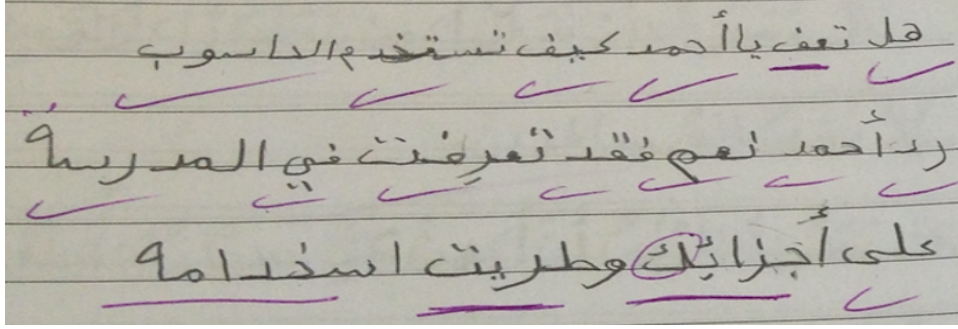


Figure 3.3: Example of a handwritten text written by a girl with dyslexia.

3.5.2 Categorisation

After reviewing several examples of text from the three resources (homework, parent, and form), it was found that the documents contained words, sentences, and paragraphs. Consequently, for the purposes of the transcription of the text into a spreadsheet, the element 'Category' was included, in order to classify the types of textual data into words, sentences, and paragraphs. Figure 3.4 shows the number of documents obtained from each category.

3.6 Participant Information

This study involved 904 students with dyslexia who were aged between eight and 13 years. The majority age of the participants was nine years. Figure 3.5 illustrates the age range of the participants.

While the participants included both male and female students, since Saudi

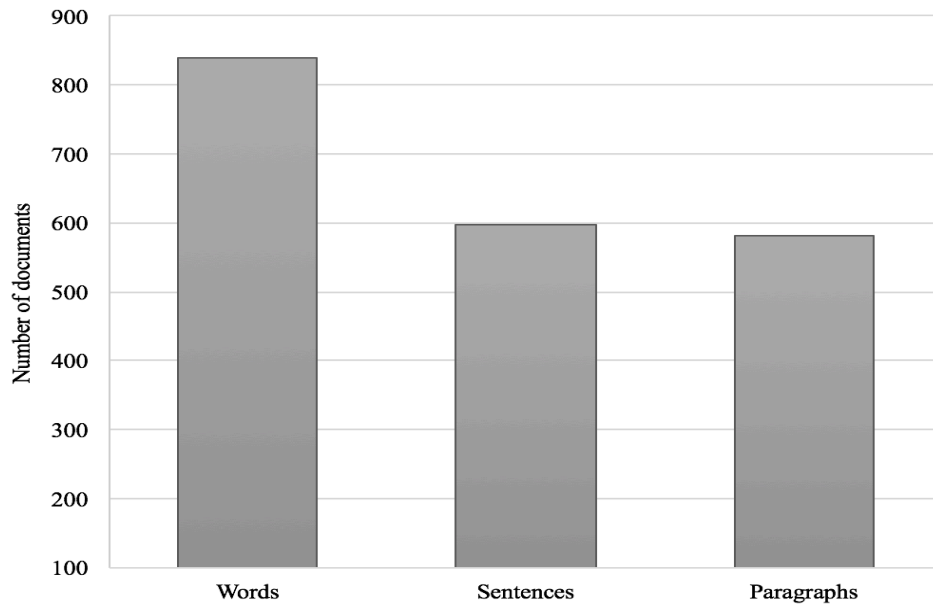


Figure 3.4: Number of documents from each category.

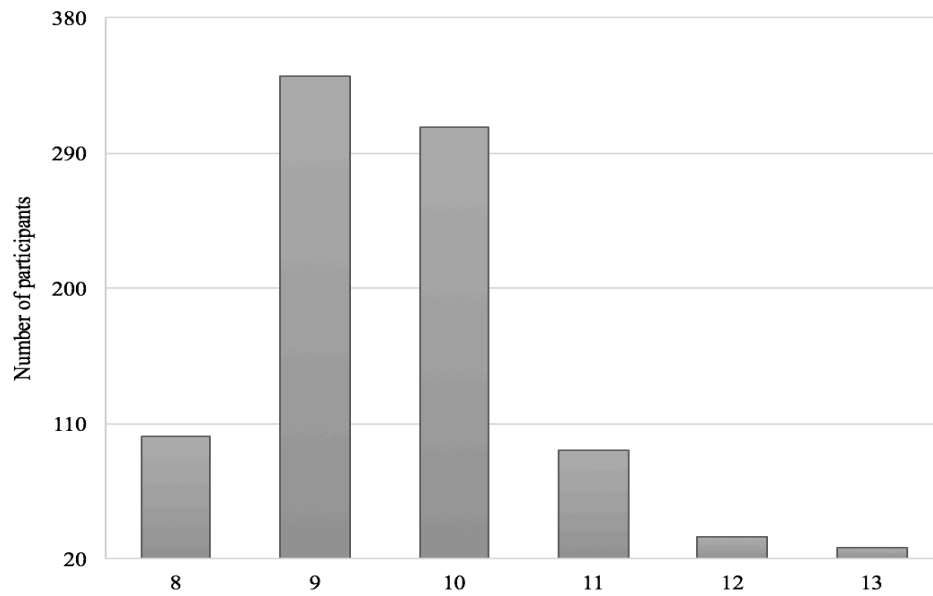


Figure 3.5: Age range of the participants.

government policy segregates the genders in education and other fields, the researcher visited only the female schools, and recruited her brother to collect the texts on her behalf from male schools with a resource room. Con-

sequently, the majority of the texts involved in this study were written by females (97% female and 3% male).

3.7 BDAC Frequency Profiling

When creating a corpus, it is necessary to provide a frequency-sorted word list (Baron et al., 2009), since word frequency may be a central premise of corpus analysis (Baker, 2006). A frequency list was therefore created to illustrate the amount of times every word and characters occurred in the text involved in this study. The data collected for the BDAC corpus comprised of 28,203 words and 154,637 characters. However, the texts differed in length, with the average of 12 words, within a range of 1-135 words. Table 3.2 illustrates the distribution profile for the top 50 word occurrences in the data.

Table 3.2: Word frequency distribution.

Freq.	Word	Freq.	Word	Freq.	Word	Freq.	Word	Freq.	Word
721	في	91	ما	61	الناس	48	بها	38	وفي
623	من	87	قال	59	المدرسة	47	محمد	38	قالت
372	و	78	مع	57	عليه	45	بين	38	لي
261	على	77	يا	57	أو	42	به	38	بعد
248	إلى	73	علا	55	أنا	42	عمر	38	المعلم
245	أن	67	ثم	54	كان	42	الى	38	أحب
182	الله	66	ولا	51	فيها	41	إن	37	يوم
142	كل	63	له	50	حتى	40	وهو	37	هو
108	عن	63	فيه	49	وقد	40	الوطن	37	بي
98	لا	62	التي	49	الذي	39	العلم	36	التي

As shown in Table 3.2 the word that appeared most frequently was “في” [E: “in”], followed by the word “من” [E: “from”], and then the “و” [E: “and”].

The top 50 words also contained errors, such as “علا” [E: “on”] and “التي” [E: “which”].

In order to investigate the number of times that each character has occurred in a corpus, the following equation can be used to calculate each character’s percentage as follow:

$$Character (\%) = \frac{\text{number of times each character has occurred}}{\text{total number of characters}} \times 100. \quad (3.1)$$

For example, in the following dyslexia sentence “أكل الولد التفاحه”, which comprises three words and 15 characters, the character ‘ا’ occurs four times while the character ‘ت’ occurs once. Therefore, the character percentage of these two characters will be $(4/15) \times 100 = 26.67\%$ and $(1/15) \times 100 = 6.67\%$, respectively.

Figure 3.6 illustrates the characters present in the study, ranked in alphabetical order, together with their character percentage.

As Figure 3.6 demonstrates, characters such as ا, ل, ي, and م had the highest frequency, while the characters ا, ؤ, and ع had a significantly lower frequency.

3.8 Conclusion

This chapter discussed how the data used to create the BDAC corpus was collected, explaining that it was gathered from three different sources: schools, via a form, and from the parents of dyslexic child. The chapter also discussed

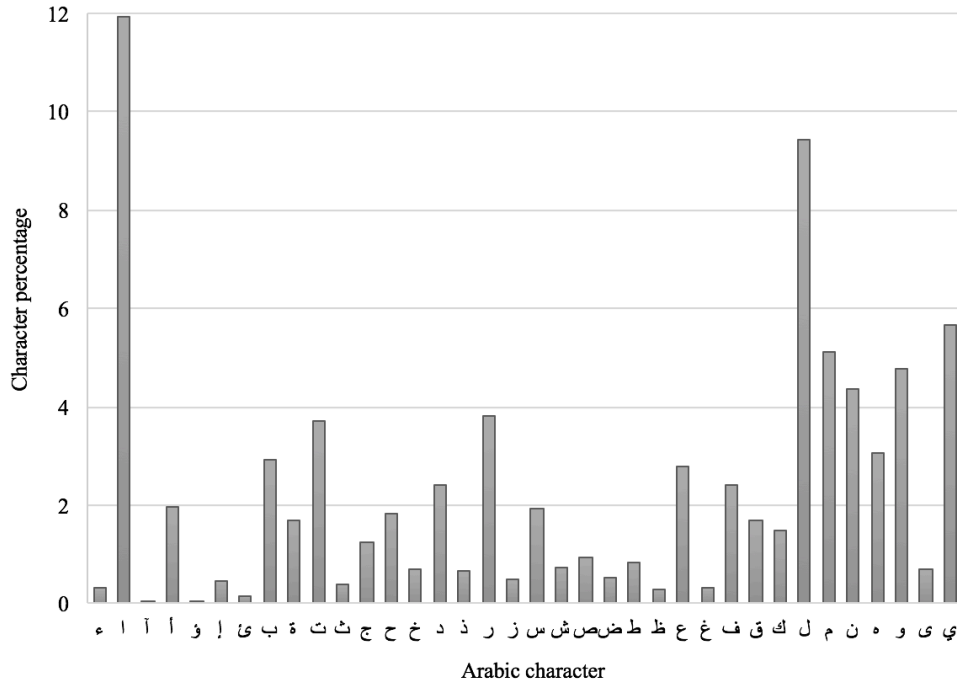


Figure 3.6: Character frequency distribution.

the procedure employed by the Saudi Ministry of Education to identify students with dyslexia in schools in Saudi Arabia.

All of the data was gathered from people who had been professionally diagnosed with dyslexia, and who were from a similar population, in terms of age and education, and with Arabic as their native language.

The chapter also described the process of transferring the text from its handwritten original to an electronic format, and provided statistical information regarding the age and gender of the participants, together with information about the text collected, including the frequency of the words and characters.

The BDAC consisted of a total of 28,203 words. At the time of writing, part of the BDAC is available for public use via the researcher's blog (ma-

haalamri.wordpress.com), and to date, four other researchers have made contact to request permission to use it in their research, thus demonstrating the potential of the corpus to serve as a platform for other researchers to build upon.

Chapter 4

Error Annotation for Dyslexic texts in Arabic

4.1 Introduction

In order to annotate the dyslexia errors in the BDAC corpus, it was crucial to select a classification scheme for annotating the errors. As far as the researcher is aware, there is no standard error classification for Arabic dyslexia errors and no classification scheme for annotating Arabic dyslexia corpora.

Therefore, this chapter concerns the development of a new classification scheme of errors made by Arabic writers with dyslexia, for use in the annotation of Arabic dyslexic text. This dyslexic error classification scheme for Arabic texts (DECA) is comprised of a list of spelling errors extracted from the previous studies discussed in the literature review (Section 2.4.2) which provide a platform for understanding and analysing specific errors

committed by writers with dyslexia.

This chapter is organised as follows: Section 4.2 discusses the basis of the classification scheme for Arabic dyslexia text, while Section 4.3 describes the dyslexic error classification scheme for Arabic texts used to annotate Arabic dyslexia errors. Section 4.4 contains an evaluation of the DECA, and Section 4.5 explains the annotation process, followed by the analysis of dyslexic errors in Section 4.6.

Aspects of the study presented in this chapter were published in the Proceedings of the Third Arabic Natural Language Processing Workshop, Association for Computational Linguistics (Alamri and Teahan, 2017).

4.2 Basis of Dyslexic Error Classification Scheme for Arabic Texts

The scheme developed for this study relies on the findings of the studies mentioned in Chapter 2 (Section 2.4.2), that discussed and analysed dyslexia errors from different aspects.

The Error Analysis in Spelling form executed by Abunayyan (2003) was selected because when the researcher conducted fieldwork in Saudi Arabia, it was found that some teachers used the form to classify their dyslexic students' errors. In addition, since teachers in Saudi Arabia were the main source of information for this study, it was necessary to locate an existing study that employed the viewpoint of teachers to classify dyslexia errors. The study conducted by Hamadneh et al. (2014) fulfilled this requirement. Meanwhile, Ali (2011) addressed many of the concepts concerning people with learning difficulties, and the types of difficulty they encoun-

tered, while Abu-Rabia and Taha (2004) compared Arabic dyslexic errors with other groups, such as age-matched group, in order to assess the existence of particular spelling error patterns among Arabic dyslexics that differ between reading-level-matched, and age-matched groups. Finally, a previous study was conducted by the researcher of the current study who analysed dyslexia errors based on texts collected from children with dyslexia (Alamri, 2013).

The DECA classification sought to remedy the shortcomings of the extant studies by combining them into a single classification under different categories, with easily comprehensible broad classes, or categories, and a comprehensive list of error types. This combination process was conducted following an assessment of the similarities and differences between the errors types noted in Chapter 2 (Section 2.4.2). Table 4.1 provides a summary of the errors located in the extant studies. The numbers in the first row refer to **1**: Alamri (2013), **2**: Abunayyan (2003), **3**: Abu-Rabia and Taha (2004), **4**: Ali (2011) and **5**: Hamadneh et al. (2014).

Table 4.1: Error types recorded in Arabic studies.

Error Type	1	2	3	4	5
Hamza on Line	x	x	x		x
Alif Hamza Above	x	x	x		x
Alif Hamza Below	x		x		x
Ya Hamza Above	x	x	x		x
Waw Hamza Above	x	x	x		x
Alif Madd	x	x	x		x
Waw Madd	x	x	x		x

Continued on the next page

Ya Madd	x	x	x		x
Confusion in Tah and Tah Marbuta/Ha	x	x			x
Confusion in Hah and Tah Marbuta	x	x			x
Confusion in Alif and Alif Maksura		x			x
Confusion in Dha and Tha	x	x			x
Confusion in similar letters			x		x
N in Tanwin	x	x	x		x
W in Damma	x	x	x		x
Y in Kasra	x	x	x		x
Omission	x	x	x	x	x
Addition	x	x	x	x	
Substitution	x	x	x	x	x
Transposition	x	x	x	x	
Different Graphemes, Same Phonetics			x	x	
Local language		x	x		
Writing a word that is similar in meaning		x			
Mirror		x		x	
Left to Right		x		x	
Sun Letter		x	x		x
Adding letter (L) to word start with letter (AL)					x
Alif Fariqa					x
Lakn, Hada, Alldhy and Allty		x	x		x
Letter written but not pronounced or vice versus					x

As shown in the above Table 4.1, there was an apparent consensus among the researchers regarding certain types of errors made by people with dyslexia,

such as ‘omission’, ‘Hamza’ and ‘Almadd’.

The findings of the extant studies regarding certain Arabic dyslexia errors, such as ‘Addition’, were consistent with the results of studies in other languages, including English (Pedler, 2007), Spanish (Rello, 2014) and German (Rauschenberger et al., 2016). The researcher therefore investigated the dyslexia errors in other languages that can be applied to Arabic error types, and consequently, two types emerged, namely word boundary errors, and multi errors.

Most of the types in the resulting classification concerned unique specificities of the Arabic language, since the system of Arabic writing contains characteristics, such as diacritics, that do not exist in other languages. However, some types in the classification, such as substitution and word boundary errors, do occur in other languages.

4.3 Dyslexic Error Classification Scheme for Arabic Texts

After investigated the dyslexia spelling errors in Section 4.2, the researcher grouped the errors into types and categories. The category is more general than the type, as it specifies where the error occurs, for example, in the *Hamza*, or in the *Almadd*. Each error category is further subdivided into a variable number of error types. Table 4.2 provides a list of the nine error categories: *Hamza*, *Almadd*, *Confusion*, *Diacritics*, *Form*, *Common error*, *Differences*, *Written Method*, *Letters Written but Not Pronounced (or vice versa)*, together with the *Other* category. The first version of the classification contained 35 error types within these categories, including *Other*,

which was added if the error present was not represented by any of the categories.

Alfaifi (2015) suggested that a two character tag can be used to represent the error, the first specifying the category, and the second specifying the error type. For example, for “الهمزة على الألف” [E: “Alif Hamza Above”], the tag was <HA>; in which ‘H’ indicated the category “الهمزات” [E: “Hamza”], and ‘A’ indicated the error types “الألف” [E: “Alif Above”]. As a further illustration, if the erroneous word was “ثيَمر”, and the intended word was “ثَمَار” [E: “fruits”], thus the author wrote ‘يَ’ [E: ‘Y’] , instead of the diacritical ‘َ’ [E: “Kasra”], and omitted the letter ‘أ’ [E: “A”], where the erroneous word had one wrong letter added in one location, and a correct letter missing in another location. Therefore, to indicate the two different types of errors, the underscore symbol was employed between the tags, as follows: <DY_AA>, with <DY> indicating the use of ‘يَ’ [E: ‘Y’] instead of ‘َ’ [E: “Kasra”] (see row 19 in Table 4.2), and <AA> indicating the error in “مد الألف” [E: “Alif Madd”] (see row 7 in Table 4.2).

Table 4.2: Version 1 of Dyslexic Error Classification Scheme for Arabic Texts (DECA).

الرمز - Tag	أنواع الأخطاء - Error Type	الفئة - Category
<HH>	الهمزة على السطر - Hamza on Line	الهمزات Hamza
<HA>	الهمزة على الألف - Alif Hamza Above	
<HB>	الهمزة تحت الألف - Alif Hamza Below	
<HY>	الهمزة على الياء - Ya Hamza Above	
<HW>	الهمزة على الواو - Waw Hamza Above	
<OT>	لم تذكر في الهمزات - Other	
<AA>	مد الألف - Alif Madd	المدود Almadd
<AW>	مد الواو - Waw Madd	
<AY>	مد الياء - Ya Madd	
<TM>	لم تذكر في المدود - Other	
<CT>	بين التاء المفتوحة والتاء المربوطة - Confusion in Tah and Tah Marbuta/Hah	الخلط Confusion
<CH>	بين الهاء والتاء المربوطة - Confusion in Hah and Tah Marbuta	
<CA>	بين الألف الممدودة والألف المقصورة - Confusion in Alif and Alif Maksura	
<CD>	بين الظاء والضاد - Confusion in Dha and Tha	
<CV>	الخلط بين حروف متشابهة - Confusion in similar letters	
<OM>	لم تذكر في الخلط - Other	
<DN>	نون مكان التنوين - N in Tanwin	الحركات Diacritics
<DW>	واو مكان الضمة - W in Damma	
<DY>	ياء مكان الكسرة - Y in Kasra	
<OD>	لم تذكر في الحركات - Other	
<FW>	وصل ماحقه الفصل من الحروف او فصل ماحقه الوصل من الحروف - Word boundary errors	شكل الكلمة Form
<FM>	أخطاء متعددة - Multi Errors	
<OF>	لم تذكر في الشكل - Other	
<MO>	حذف - Omission	الاطفاء الشائعة Common errors
<MA>	إضافة - Addition	
<MS>	تبديل - Substitution	
<MT>	تحويل - Transposition	
<DD>	عدم القدرة على تفریق بين حروف متشابهة لفظاً مختلفة شكلاً - Different Graphemes, Same Phonetics	الاختلافات Differences
<DI>	كتابة بناء على اللهجة المحلية - Local language	
<DS>	كتابة كلمة متشابهة للمعنى - Writing a word that is Similar to the Meaning	
<OI>	لم تذكر في الاختلافات - Other	
<WM>	مرآة - Mirror	طريقة الكتابة Writing method
<WL>	كتابة من اليسار لليمين - Left to Right	
<OW>	لم تذكر في الكتابة - Other	
<LS>	لام الشمسية - Sun Letter	حروف تكتب ولا تنطق او العكس Letter written but not pronounced or vice versa
<LM>	إدخال اللام على ما فيه (ال) - Adding letter (L) to words start with letter (AL)	
<LA>	ألف بعد واو الجماعة - Alif Fariqa	
<LL>	(لاكن - لاكتها ...) - (Lakn ...)	
<LH>	(هاذا - هاذ - هاذان) - (Hada ...)	
<LT>	(التي) - (Alty)	
<LD>	(الذي) - (Alldhy)	
<LK>	(ذالك - بذلك ...) - (Dahlk ...)	
	لم تذكر في حروف - Other	
<OT>	لم تذكر في أي مجموعة - Other	اخرى - Other

4.4 Evaluating the DECA

In order to evaluate the comprehensiveness, appropriateness, and clarity of the classification scheme employed for the DECA, and to determine whether it was suitable for the purpose of error analysis, three evaluation experiments were undertaken. The first was an annotation sample of the BDAC, the second an evaluation conducted by specialists in the field of learning difficulties, and the third used inter-annotation agreement.

4.4.1 Annotation Sample Evaluation

According to Pustejovsky and Stubbs (2012) when conducting the first round of annotations, it is best to select a sample of the corpus to annotate, in order to assess how well the annotation task works in practice. Therefore, a 500 word sample was randomly selected from the BDAC for this evaluation. Two annotators, ‘A1’ and ‘A2’, participated, the former is an Arabic language lecturer, while the latter held a bachelor’s degree in Computer Science and is a native Arabic speaker.

Each annotator was given the same sample of 500 words, with the errors in the sample identified in advance, and was instructed to add the most appropriate tag that matched the error type, using classification Version 1 (see Table 4.2) to annotate all of the errors. The annotators were then asked to provide a list of the types of errors they encountered that matched the classification, and to indicate whether there were any types not listed in the classification. Thus, the purpose of this evaluation was to assess the clarity of the error tags, and to determine whether any were absent from the classification.

Both A1 and A2 stated that all the types in Table 4.2 were clear, but while A2 did not find any error types that were not from the classification, A1 suggested that the following should be added: “تكرار الحروف” [E: “Repeated Letters”], and “عدم القدرة على التفريق بين شكل الحرف إذا كان في بداية او وسط او نهاية الكلمة” [E: “Written Form in Beginning, Middle or End”]. A1 suggested that these should be added because they were found in the sample in words such as “اللقمة” [E: “the bite”], in which the letter ‘ل’ [E: “L”] was repeated three times, and “أبوها” [E: “her father”], in which the letter ‘هـ’ [E: “h”] was written in the middle of the word, rather than spelled correctly as “أبوها”. Version 1 was subsequently edited to include these two types, and Version 2 (see Table 4.3) of the classification therefore contained nine categories and 37 types.

Table 4.3: Version 2 of Dyslexic Error Classification Scheme for Arabic Texts (DECA).

الرمز - Tag	أنواع الأخطاء - Error Type	الفئة - Category
<HH>	الهمزة على السطر - Hamza on Line	الهمزات Hamza
<HA>	الهمزة على الألف - Alif Hamza Above	
<HB>	الهمزة تحت الألف - Alif Hamza Below	
<HY>	الهمزة على الياء - Ya Hamza Above	
<HW>	الهمزة على الواو - Waw Hamza Above	
<OT>	لم تذكر في الهمزات - Other	
<AA>	مد الألف - Alif Madd	المدود Almadd
<AW>	مد الواو - Waw Madd	
<AY>	مد الياء - Ya Madd	
<TM>	لم تذكر في المدود - Other	
<CT>	بين التاء المفتوحة والتاء المربوطة - Confusion in Tah and Tah Marbuta/Hah	الخلط Confusion
<CH>	بين الهاء والتاء المربوطة - Confusion in Hah and Tah Marbuta	
<CA>	بين الألف الممدودة والألف المقصورة - Confusion in Alif and Alif Maksura	
<CD>	بين الظاء والضاد - Confusion in Dha and Tha	
<CV>	الخلط بين حروف متشابهة - Confusion in similar letters	
<OM>	لم تذكر في الخلط - Other	
<DN>	نون مكان التنوين - N in Tanwin	الحركات Diacritics
<DW>	واو مكان الضمة - W in Damma	
<DY>	ياء مكان الكسرة - Y in Kasra	
<OD>	لم تذكر في الحركات - Other	
<FW>	وصل ماحقه الفصل من الحروف او فصل ماحقه الوصل من الحروف - Word boundary errors	شكل الكلمة Form
<FM>	أخطاء متعددة - Multi Errors	
<FR>	تكرار الحروف - Repeated Letters	
<OF>	لم تذكر في الشكل - Other	
<MO>	حذف - Omission	الايخطاء الشائعة Common errors
<MA>	اضافة - Addition	
<MS>	تبديل - Substitution	
<MT>	تحويل - Transposition	
<DD>	عدم القدرة على التفريق بين حروف متشابهة لفظا مختلفة شكلا - Different Graphemes, Same Phonetics	الاختلافات Differences
<DF>	عدم القدرة على التفريق بين شكل الحرف إذا كان في بداية الكلمة أو وسطه أو نهايته - Form of the letter in the Beginning, Middle or End	
<DI>	كتابة بناء على اللهجة المحلية - Local language	
<DS>	كتابة كلمة مشابهة للمعنى - Writing a word that is Similar to the Meaning	
<OI>	لم تذكر في الاختلافات - Other	
<WM>	مرآة - Mirror	طريقة الكتابة Writing method
<WL>	كتابة من اليسار لليمين - Left to Right	
<OW>	لم تذكر في الكتابة - Other	
<LS>	لام الشمسية - Sun Letter	حروف تكتب ولا تنطق او العكس Letter written but not pronounced or vice versa
<LM>	دخول اللام على ماقيه (ال) - (AL) - Adding letter (L) to words start with letter (AL)	
<LA>	ألف بعد واو الجماعة - Alif Fariqa	
<LL>	(لاكن - لاكنها ...) - (Lakn ...)	
<LH>	(هاذا - هاذ - هاذان) - (Hada ...)	
<LT>	(التي) - (Ally)	
<LD>	(الذي) - (Alldhy)	
<LK>	(ذلك - بنلك ...) - (Dahlk ...)	
	لم تذكر في حروف - Other	
<OT>	لم تذكر في أي مجموعة - Other	الخرى - Other

4.4.2 Teachers' Feedback Evaluation

The second evaluation involved a questionnaire (Appendix 6) that was sent to two evaluators who had agreed to participate. These evaluators are primary school teachers of children with learning difficulties, who were referred to as T1 and T2. They were given Version 2 of the DECA, and were asked to select the appropriate tags to evaluate how easily they were identified, the participants were asked to provide feedback regarding whether they felt the list included all of the errors made by students with dyslexia, and whether the categories were appropriate.

Both of the evaluators were able to locate the correct tag for all of the sentences (see Appendix 6), with the exception of that containing the error word “التي” [E: “which”], as T1 chose the <FR> tag, rather than <LT>. Both of the evaluators believed the tags to be appropriately named. Meanwhile, T2 found that the first and fourth sentences were ‘Very suitable’, but that the second, third and fifth sentences were only ‘Suitable’, and T1 found that all of the sentences were ‘Suitable’. They agreed regarding how easy they found it was to locate the appropriate tag for the second and third sentences, while T1 found it ‘Easy’ for the first and fourth sentences, and ‘Difficult’ for the last sentence. T2 found it ‘Very easy’ for all of the sentences except the second sentence, which was ‘Easy’. In addition, the evaluators agreed that the table of the types of dyslexia errors was comprehensive.

4.4.3 Inter-annotator Agreement Evaluation

The purpose of the third evaluation was to measure the inter-annotator agreement of the DECA when applied to a sample of text, by assessing to what extent the annotators assigned the same tag to the errors. For the

purpose of this evaluation, a sample of 1,000 words was provided to two of the annotators (A2 and T1) from previous evaluations (Section 4.4.1 and 4.4.2), as well as to the researcher of this study (N1). Each misspelling in the sample was marked, and the intended word was provided, then each annotator was required to add the most appropriate tag for the error type, using the DECA version 2.

In order to analyse the reliability of the agreement among the annotators, Kappa statistics were utilised. Kappa was originally proposed by Cohen (1960), and was deemed to be the most appropriate statistical measure for measuring and comparing the agreement between annotators when classifying a text into types. When the Kappa value is above 0.6, it demonstrates reasonable agreement between two elements, while a value between 0.8 and 1 represents an excellent agreement (Landis and Koch, 1977).

The results of the analysis revealed that the agreement between T1 and N1 was 87%, with a weighted Cohen’s Kappa value of ($k = 0.87$). The annotators A2 and N1 had the highest agreement (88%), with a weighted Cohen’s Kappa value of ($k = 0.86$). In contrast, A2 and T1 had 84% agreement, resulting in a weighted Cohen’s Kappa value of ($k = 0.81$). These results show a high agreement between annotators.

4.5 The Annotation Process of the BDAC Corpus

As Granger (2003) noted, while error annotation is an exceedingly tedious task that must be undertaken with care, it is immensely important, as it enables a researcher to gain quick access to particular error statistics.

Figure 4.1 presents an interface of the tool that was created to facilitate a rapid annotation process for this study. In order to enable the retrieval of corpus texts, the tool was integrated into the BDAC database on Microsoft Excel. The tool was developed according to a process proposed by Pustejovsky and Stubbs (2012), known as ‘stand-off annotation’, which converts (tokenizes) the text into tokens based on whitespaces, and thereby enables an annotator to select the tags of the error tokens more easily. Each token was located on a separate line, and the erroneous words were annotated alongside each type of error, based on the DECA classification version 2 and the correct spelling of the erroneous word. The tool’s interface was provided in both Arabic and English, as shown in Figure 4.1

Figure 4.1 illustrates the process employed when an annotator has chosen a word to annotate. The “النص الاصيل” [E: “Raw Text”] and the “النص الصحيح” [E: “Correct Text”] are described Chapter 3 in Section 3.4. In the example, the error is located in Token 2, hence the annotator selects ‘Token 2’, as it is an error word, by double-clicking on the error (Token 2) in the text area labelled “النص الاصيل” [E: “Raw Text”], then selects the correct word from the text area labelled “النص الصحيح” [E: “Correct Text”], again by double-clicking. The appropriate tag is then selected from the list. The option “تنفيذ” [E: “Apply”] is then clicked, with it appearing in the “النص الاصيل” [E: “Raw Text”] area, in order that the annotator can see the annotation of the errors in this area before saving the annotation.

The procedure is repeated with each error found in the text. In the case of a word that contains more than one type of error, as denoted by Token 6, the annotator can add another tag via the “+” button, and choose another tag, which is separated by ‘_’. As a result, the annotation for Token 6 is:

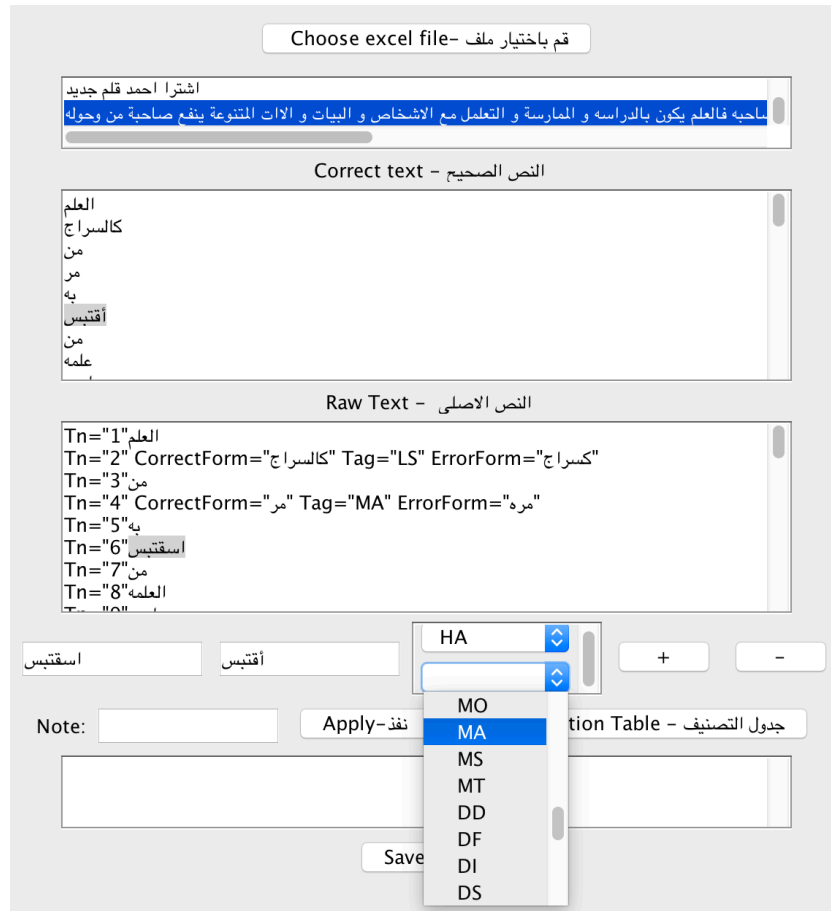


Figure 4.1: Interface of the tool developed to facilitate the annotation process.

Tn="6" CorrectForm="أقتبس" Tag="HA_MA" ErrorForm="استقتبس"

Each error token requires two annotations: one for the correct word, and another for the error type, as follows:

Tn="1" CorrectForm="الشمس" Tag="LS" ErrorForm="اشمس"

where:

- **Tn** = Token number (position of the word within the text);

- **CorrectForm** = The intended word of the error word;
- **Tag** = Contains an abbreviation of the error type from DECA version 2;
- **ErrorForm** = The error word.

The BDAC corpus was fully annotated using DECA version 2, using this tool, and the combined information was ultimately converted to an XML file, as shown in Figure 4.2.

4.6 Analysis of Dyslexic Errors in the BDAC

The annotation of a corpus provides a significant advantage in terms of enabling a search for particular error types, or groups of errors, in exactly the same way that individual words can be searched (Nicholls, 2003). Once this study’s annotation was conducted, the corpus analysis became the simple process of extracting the tags, or errors, and their corresponding target word.

The BDAC corpus comprised more than 8000 errors. In some cases these errors contains more than one error type. Moreover, some of the errors were found to occur more than others in the corpus. The highest error occurrence was in “علا”, the correct form of which is “على” [E: “On”]. The error type concerned is (CA), which is located under the “الخلط” [E: “Confusion”] category. This was followed by “التي” [E: “which”] that falls under the “حروف تكتب ولا تنطق أو العكس” [E: “Letter written but not pronounced or vice versa”] category.

The highest number of errors for a specific category was for the “الأخطاء الشائعة”


```

<?xml version="1.0" encoding="UTF-8"?>
<BDAC>
<Record_info>
<Participant_info>
<Age>9<\Age>
<Gender>Female - انثى <\Gender>
<\Participant_info>
<Text_info>
<TextSource>HW - الواجب <\TextSource>
<Text n="1">
<Category>sentence - جملة <\Category>
<RawText>ذهبت هندون الى الحديقة<\RawText>
<CorrectText>ذهبت هند إلى الحديقة<\CorrectText>
<Error_analysis>
<Token Tn="1">ذهبت<\Token>
<Token Tn="2" CorrectForm="هند" Tag="DN">هندون<\Token>
<Token Tn="3" CorrectForm="إلى" Tag="HB">الى<\Token>
<Token Tn="4" CorrectForm="الحديقة" Tag="CH">الحديقة<\Token>
<\Error_analysis >
<\Text>
<\Text_info>
<\Record_info>
<\BDAC>

```

Figure 4.2: XML sample with stand-off annotation by tokens.

[E: “Common errors”] category, with 2,717 instances, followed by 1,621 errors in the “الهمزات” [E: “Hamza”] category, and 1,553 errors in the “الخلط”

[E: “Confusion”] category. Meanwhile, the lowest two types of errors fell within the “الاختلافات” [E: “Differences”], and “شكل الكلمة” [E: “Form”] categories.

Furthermore, people with dyslexia are commonly known to confuse “بين التاء” [E: “Tah and Tah Marbuta/Hah”], and in this study, 523 instances of errors fell under this type. Meanwhile, the majority of the extant studies agreed that people with dyslexia find diacritics marks difficult, and this study found that the highest number of errors fell under “نون مكان التنوين” [E: “N in Tanwin”], with 309 instances. In addition, a small number of errors fell under the “حروف تكتب ولا تنطق أو العكس” [E: “Letter written but not pronounced or vice versa”] category, the most frequent of which was in the “اللام الشمسية” [E: “Sun-letters”].

This study also evaluated the position of letters in terms of whether the shape of the letter differed depending on position if it is in the beginning, middle, or the end of a word. For example, ‘ة’ [E: “Tah Marbuta”] was written in the middle of a word, such as “منازلةم”, instead of “منازلهم” [E: “their homes”], or whether a letter was written at the beginning of a word, when it should be at the end, such as ‘حـ’, instead of ‘ح’, in the word “فلاح”, the correct form is “فلاح” [E: “Farmer”].

The analysis also revealed that a space was sometimes located incorrectly (145 instances), and this error type is under the “وصل ماحقه الفصل من الحروف أو” [E: “Word boundary errors”] category. Moreover, it was noticed that the majority of these incorrectly-located spaces were either before the suffix, or after the prefix, such as “لل تخلص”, instead of “للتخلص” [E: “to get rid”], in which the space was after the prefix ‘ل’.

In some instances, two letters were written to represent one letter, such as

“علاء”, instead of “على” [E: “on”], where ‘ا’ and ‘ء’ were written to represent ‘ى’, in a type of error known as “أخطاء متعددة” [E: “Multi errors”], under the category “شكل الكلمة” [E: “Form”].

Moreover, the new error classification scheme was also applied to reanalyse the errors taken from the researcher’s previous work. The result shows that the highest number of errors was located in “المدود” [E: “Almadd”] category with total number of 255 instances, followed by “الهمزات” [E: “Hamza”] category with 193 instances where the lowest categories were “شكل الكلمة” [E: “Form”] and “الاختلافات” [E: “Differences”] categories.

4.7 Conclusion

This chapter introduced the DECA, which was developed to facilitate the annotating of dyslexia errors in Arabic, explaining that the current version of the DECA includes 37 types of errors classified under nine categories.

Five people evaluated the DECA, using it to annotate the dyslexia errors in a sample of text from the BDAC. The following three evaluations were conducted, in order to assess the DECA’s reliability and effectiveness: The first evaluation determined whether the error tags were sufficiently clear, and assessed whether any new types were not listed in the DECA. The second evaluation assessed whether the DECA included all of the errors committed by students with dyslexia, and whether the categories were appropriate, and the third evaluation measured the inter-annotator agreement of the DECA when it was applied to a sample of the BDAC revealing that there is a high agreement between annotators.

The outcomes of the evaluations discussed in this chapter were positive, sup-

porting the fact that the DECA represents a complete classification framework for dyslexia errors in Arabic. It can be claimed that it therefore represents the most comprehensive classification for dyslexia error in Arabic, and has the potential to provide assistance in the field of dyslexia, computer science, and pedagogy, as it might provide a basis for improved aid for the target group.

This chapter also discusses the error annotation process, and provides an analysis of the errors that were found in the BDAC corpus.

Chapter 5

Distinguishing Dyslexic Text from Non-dyslexic Text

5.1 Introduction

A growing body of research has now begun to identify disorders using natural language processing and machine learning techniques, for example, Alzheimer’s (Thomas et al., 2005), autism (Liu et al., 2013) and depression (Zhou et al., 2015). Thus, this chapter investigates whether the text compression scheme (PPM) can be applied to the binary classification problem of distinguishing dyslexic text from non-dyslexic text, which involves identifying whether or not a text has been written by writers with dyslexia.

This chapter first explains how PPM can be used to classify dyslexia text in Section 5.1.1. Section 5.2 describes the dyslexia and non-dyslexia corpora used in the experiments. Section 2.6.1 explains the evaluation techniques

followed by a number of experiments that were performed to classify the dyslexia text in Sections 5.3. The chapter finishes with a comparison of the performance of a PPM compression-based text classifier on the problem of classification of dyslexic and non-dyslexic Arabic text with that of standard feature-based classifiers, such as Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM) in Section 5.3.3.

5.1.1 Example of Dyslexia Classification

Concerning the dyslexia classification method, separate texts were selected as representative of dyslexic text and non-dyslexic text, in order to train the PPM character-based language models. The main intention when utilising PPM compression techniques for classification of texts is to draw on their capacity to generate an accurate language model. Each of the texts were compressed using static PPMD character models trained using the representative texts.

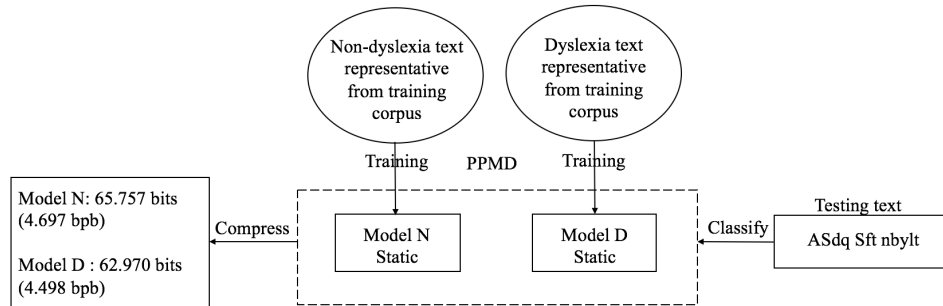


Figure 5.1: Dyslexia classification using PPMD.

Figure. 5.1 shows how the dyslexia model (Model D) trained from dyslexia texts and the non-dyslexia model (Model N) trained from non-dyslexia texts are used to compute codelength ratios by compressing the test file based on the training of the two models. The sample input text “اصدق صفت نبيلت” [E:

“Honesty is a worthy attribute” B: “<ASdq Sft nbylt” R: “aşdq şft nbylt”] was written by a writer with dyslexia. The results show the dyslexia model requires 4.49 bits per byte calculated by equation (2.15) to compress the text whereas the non-dyslexia model requires 4.69 bits per byte (where ‘bits per byte’ is the compression ratio and is measured in compressed output size in bits divided by original text size in bytes). As a result, the test text is classified as dyslexic text (calculated by equation 2.16) and therefore has been classified correctly.

5.2 Text Corpora

This section describes the text corpora that have been used in the experiments described below in Section 5.3. Table 5.1 provides abbreviations of the text corpora, their references and the number of words and characters in each corpus. A more detailed description of each of these corpora is provided below. Examples of each corpora are also provided in Table 5.2.

Table 5.1: Text corpora used in the experiments.

Abbrev.	Corpus & Ref.	Words	Characters
ALC	Arabic Learner Corpus (Alfaifi, 2015)	282,732	1,533,919
BDAC	Bangor Dyslexia Arabic Corpus [Chapter 3]	28,203	154,637
BNDAC	Bangor Non-Dyslexia Arabic Corpus [This chapter]	9,099	49,515

The Arabic Learner Corpus (ALC) This corpus includes written and spoken data from learners of Arabic in Saudi Arabia, between 16 and 25 years old. Table 5.2 lists a sample of text taken from the corpus in the first row.

The Bangor Dyslexia Arabic Corpus (BDAC) The description of

BDAC is provided in Chapter 3. An example of a text written by a person with dyslexia is shown in Table 5.2.

The Bangor Non-Dyslexia Arabic Corpus (BNDAC) This was created for the work described in this Chapter. The BNDAC has a total of 9,099 words written by 66 non-dyslexic children between the ages of 8 and 13 from male and female participants. The researcher contacted parents, who then gave permission (see Appendix 2) to use text written by their child. An example of a text written by a person without dyslexia in Arabic taken from this corpus is shown in Table 5.2. More examples are provided in Appendix 7.

Table 5.2: Some examples of text found in each of the corpora used in the experiments.

Abbrev.	Some examples of text found in the corpus
ALC	الإجازة فى السنة الماضية كانت الإجازة ممتعة جدا ذهبنا إلى مكة المكرمة أنا مع أخي المهندس وأهله اعتمرنا وكان أول ما شهدت الكعبة كان الإجازة مفيد جدا لأننى زرت مكة ورأيت بعض عباد الله
BDAC	فهذا يبذر لهذا قمحا يأكله وهذا يعمل لهذا ثوبا يلبسه وهذا يصنعه لهذا بيتا يساكنه وهذا ينجز لهذا بابا يغلقه على بيته وغير ذلك مما لا يكاد يدركه العدد من الصناعات والحاجات لانه ليس في اصطاعات انسان واحد أن يكون فلاحا نساجا بناء نجارا
BNDAC	محبة الجار يحكى أن تاجرا ورث دارا عن أبيه فكان يحبها كثيرا ويحافظ عليها ولم يفكر يوما في بيعها أو هجرها ولكن لما كسدت تجارته وتراكت عليه الديونا عرضها للبيع و حدد مبلغا كبيرا من المال ثمننا له

5.3 Classification Experiments

To evaluate the quality of the PPM classification method for dyslexia classification, three experiments were conducted using the three corpora as described in Section 5.2. We also conducted additional experiments to determine which classification methods produced the best results, using two

well-known algorithms, the Multinomial Naïve Bayes and Support Vector Machines (SMO in Weka) algorithms (Bouckaert et al., 2013).

5.3.1 Dyslexia Corpus and Learner Corpus Experiment

The purpose of this initial experiment was to test the ability of the PPM model to distinguish between dyslexic and non-dyslexic text in Arabic using two corpora. The Bangor Dyslexia Arabic Corpus was selected as the dyslexic corpus and the Arabic Learner Corpus was selected as the non-dyslexic corpus. The reason why ALC was selected in this initial experiment was that the data were collected from learners of Arabic in Saudi Arabia and therefore thought to be reflective of the early learning (primary school) text collected for the BDAC.

As mentioned in Chapter 2, Section 2.7.1.4, in order to deal with Arabic text it is required to perform transliteration from the original Arabic text using the Buckwalter transformation. Some files in the ALC contains diacritical marks and punctuations marks, whereas the BDAC does not contain these marks. Therefore, these marks were removed from the ALC corpus for these experiment.

To run the experiment, a 10-fold cross-validation was performed using the data from the BDAC and ALC. The corpora were split into 10 fold each. Subsequently, each test text that was not part of the training data was compressed using PPMD character models on the remaining text, using different orders (from 0 to 5). As mentioned in Section 2.7.1.3, the best model is judged to be the one with the lowest codelength ratio. Figure 5.2 shows the accuracy (calculated by equation 2.4) of different PPMD orders.

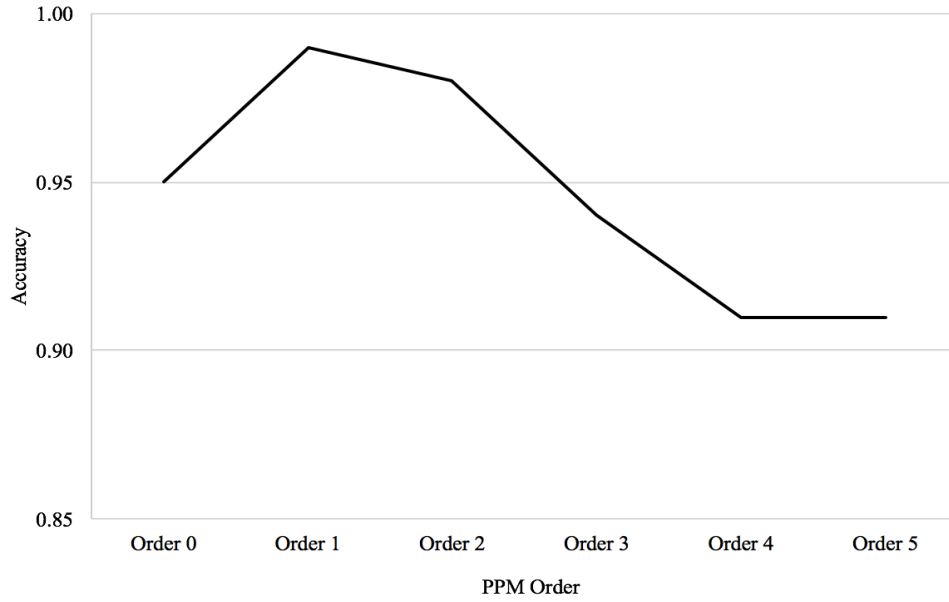


Figure 5.2: Accuracy of dyslexia (BDAC) versus non-dyslexia (ALC) text classification using different PPMD orders.

As shown in Figure 5.2, the accuracy increased from order 0 to order 1 but then decreased. The results of this experiment are shown in Table 5.3. The best F_1 score result is shown in bold font.

Table 5.3: Classification results of dyslexia (BDAC) versus non-dyslexia (ALC) text using different PPMD orders.

	Order 0	Order 1	Order 2	Order 3	Order 4	Order 5
Rec.	0.91	0.99	0.96	0.88	0.83	0.82
Prec.	0.99	0.99	1.00	0.99	0.99	1.00
F_1 score	0.95	0.99	0.97	0.94	0.91	0.90

The results shows that order 1 is the most effective at achieving a high accuracy and F_1 score.

The high accuracy raises a question whether the style or language of the text plays an important role in helping to classify the text since PPM is

an excellent classifier at language identification, genre classification, and authorship identification (Teahan and Harper, 2003; Altamimi and Teahan, 2017). For example, the ALC was written by adults learning Arabic, while the BDAC was written by children with dyslexia. Further investigations as described below were carried out to determine the validity of the preliminary results of the first experiment. Thus, the next experiment used text written by children for both the dyslexia and non-dyslexia text.

5.3.2 Arabic Children Corpora Experiment

This experiment was conducted using a non-dyslexia corpus with the same characteristics as the dyslexia corpus, such as the writer’s age and style of text. However, in order to carry out such an experiment, it was first necessary to use a non-dyslexia corpus that was constructed in a similar manner as to the dyslexia corpus (BDAC). That is, the non-dyslexia text needed to be written by children as dictated from their curriculum or by parents, but there was no prior corpus that met these requirements. Therefore, a new non-dyslexia corpus (BNDAC) was created for the purposes of this experiment. Its details are described in Section 5.2.

As mentioned in Section 5.2, the BNDAC corpus was written by 66 participants, therefore, we chose 66 participants from the BDAC corpus as well. After that, transliteration from the original Arabic text using the Buckwalter transformation was applied on both texts. To run the experiment, a 10-fold cross-validation was performed using the BDAC and BNDAC, each test text that was not part of the training data was compressed using PPMD character models on the remaining text, using different orders from 0 to 5.

Initial results using these two corpora were disappointing. Table 5.4 shows

the F_1 score of order 0 to order 5. The results obtained were poor and would not be useful as a means of accurately classifying dyslexic text. These result further supported the conjecture that the style of the corpus plays an essential role in the classification.

Table 5.4: Classification results of dyslexia (BDAC) versus non-dyslexia (BNDAC) text using different PPMD orders.

	Order 0	Order 1	Order 2	Order 3	Order 4	Order 5
Rec.	0.62	0.60	0.53	0.51	0.39	0.40
Prec.	0.66	0.76	0.83	0.80	0.76	0.72
F_1 score	0.64	0.67	0.64	0.62	0.52	0.52

As such, there is a need to improve the compression modelling based classification, possibly by improving the compression. One technique that leads to a significant improvement in compression performance is Bi-graph Substitution for PPM (“BS-PPM”) (Alhawiti, 2014) (see Figure 5.3). This is where the most commonly repeated bi-graphs in the text are replaced with new single symbols that are essentially added to an expanded alphabet. A bi-graph is the same two characters showing up together consecutively, such as “في” [E: “in”] for Arabic. The bi-graph replacement is done during the pre-processing step. The effect of the bi-graph replacement pre-processing is that fewer symbols are being processed which makes the text more predictable therefore enhancing the compression results. During the post-processing step, the new symbols are replaced with the original bi-graphs by expanding them, thereby recovering the original text in a lossless manner.

Previous studies by Teahan (1998) and by Alhawiti (2014) have shown that PPM with bi-graph replacement as a pre-processing step prior to compression usually leads to small improvements in compression for English text

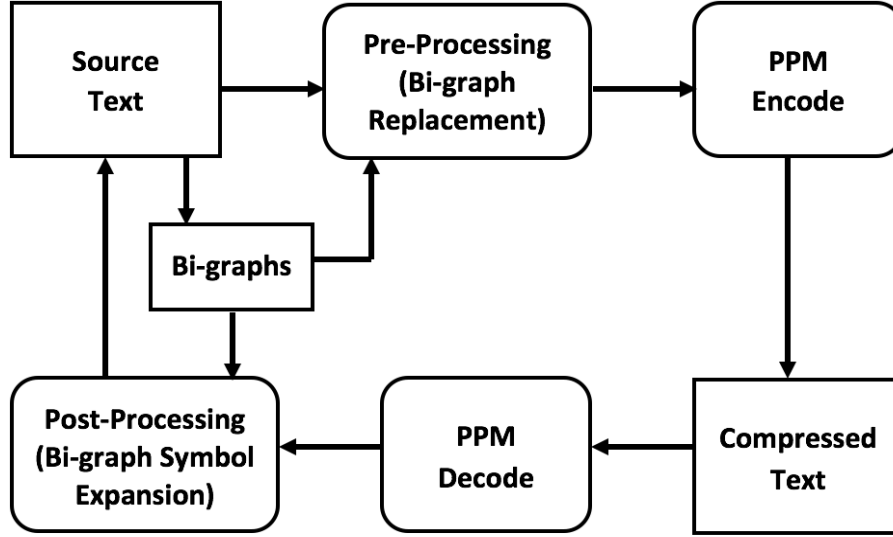


Figure 5.3: Pre-processing and post-processing for PPM using bi-graph replacement.

but produces significant improvements for Arabic text. Alhawiti (2014) conducted an experiment by examining the most frequent 20 bi-graphs for Arabic and English. The experiment used the Corpus of Contemporary Arabic (CCA) (Al-Sulaiti and Atwell, 2006), the Brown corpus (Francis and Kucera, 1979) and the LOB corpus (Johansson, 1980). The result showed that the top 20 bi-graphs take up almost 10% of the English texts and represent about 17% of the Arabic text. Moreover, Al-kazaz (2018) investigated the number of times that each n-graph occurs in a large Mixed Arabic corpus. The large Mixed Arabic corpus is a combination of the BACC (Alhawiti, 2014), Corpus A created by Alkahtani (2015) and a selection of files from the King Saud University Corpus of Classical Arabic (KSUCCA) (Alrabiah et al., 2013). Al-kazaz (2018) found that the occurrence of the top 30 bi-graphs represent 21% of the corpus.

An experiment reported here examined the top 20 bi-graphs of the BDAC

and BNDAC corpus. Each character's percentage (%) was calculated using equation 3.1 in Chapter 3. The result shows that for the BDAC, the top 20 bi-graphs takes up 15.71% while for the BNDAC, it takes up 16.65%. Table 5.5 shows the bi-graphs frequency statistics for the BDAC corpus and BNDAC corpus.

Table 5.5: Bi-graphs frequency statistics for the BDAC corpus and BNDAC corpus.

Ranking	BDAC			BNDAC		
	Bi-graphs	Freq.	%	Bi-graphs	Freq.	%
1	ال	7583	4.899	ال	2618	5.287
2	لم	1667	1.077	لم	492	0.994
3	وا	1312	0.847	وا	395	0.798
4	لا	1141	0.737	لا	376	0.759
5	من	1106	0.714	ما	344	0.695
6	في	1082	0.699	في	335	0.677
7	ها	1053	0.680	نا	331	0.668
8	ما	895	0.578	ان	324	0.654
9	ان	890	0.575	ها	310	0.626
10	عل	876	0.566	من	307	0.620
11	نا	810	0.523	عل	303	0.612
12	لي	754	0.487	لي	285	0.576
13	ات	690	0.445	با	254	0.513
14	را	664	0.429	له	244	0.493
15	له	654	0.422	لى	242	0.489
16	ار	649	0.419	اء	238	0.481
17	بي	634	0.409	ار	231	0.467
18	لى	621	0.401	را	215	0.434
19	لل	615	0.397	لل	209	0.422
20	ين	610	0.394	لا	198	0.400
Total		24306	15.714		8251	16.653

Therefore, for the purposes of this Arabic dyslexia classification experiment,

bi-graph replacement of the top 100 bi-graphs generated from the source text for the BDAC and BNDAC, because this was found to work best by Teahan (1998) and Alhawiti (2014).

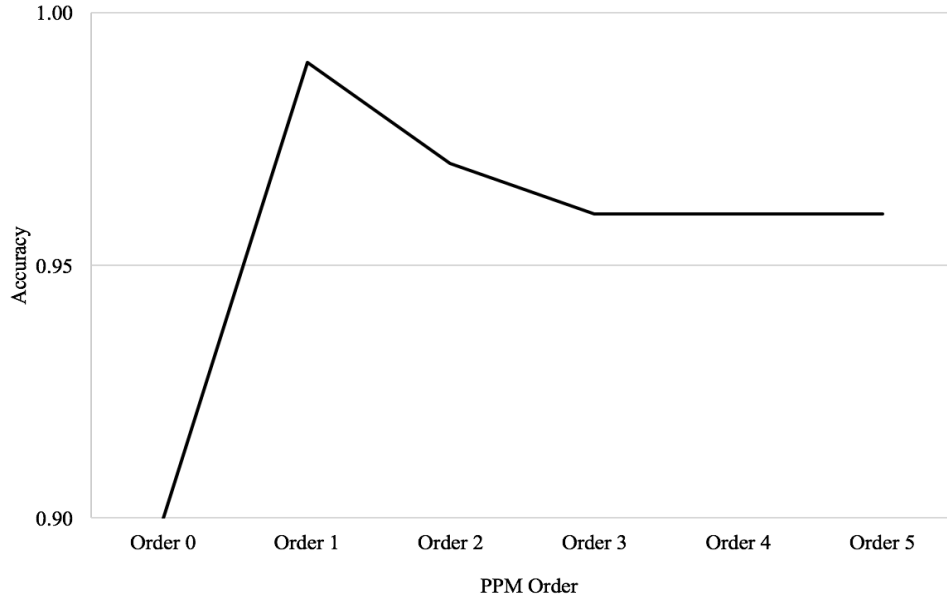


Figure 5.4: Accuracy of dyslexia (BDAC) versus non-dyslexia (BNDAC) text classification using different BS-PPM orders.

Significant improvement was achieved using the bi-graph replacement technique for the compression based classifier. Table 5.6 shows the recall, precision and F_1 score which are now improved for the different orders.

Table 5.6: Classification results of dyslexia (BDAC) versus non-dyslexia (BNDAC) text using different BS-PPM orders.

	Order 0	Order 1	Order 2	Order 3	Order 4	Order 5
Rec.	0.83	0.98	0.95	0.93	0.93	0.93
Prec.	0.96	1.00	1.00	1.00	1.00	1.00
F₁ score	0.89	0.99	0.97	0.96	0.96	0.96

Clearly, the much better compression in this case produced significantly

better classification.

Previous studies for PPM text classification have also investigated using different orders. For example, Frank et al. (2000) found order 2 is the most effective order for English topic categorisation while Almahdawi and Teahan (2018) found order 4 for emotion recognition.

In regards to dyslexia classification, reported in this thesis, a lower order was found to be more effective than a higher order model. To examine this further, Table 5.7 shows the codelength ratio for five samples documents (D1, D2, D3, D4 and D5) taken from the BDAC dyslexia corpus.

Table 5.7: Different order of BS-PPM over dyslexia and non-dyslexia corpora.

	Order 0		Order 1		Order 2	
	Dyslexia	Non-dyslexia	Dyslexia	Non-dyslexia	Dyslexia	Non-dyslexia
	Model	Model	Model	Model	Model	Model
	(bpb)	(bpb)	(bpb)	(bpb)	(bpb)	(bpb)
D1	6.55	6.27	6.05	6.33	6.09	6.60
D2	6.40	5.87	5.53	5.69	5.56	5.73
D3	6.67	6.14	5.50	5.68	5.85	5.62
D4	6.57	6.04	5.71	5.93	6.01	5.93
D5	6.60	6.38	5.90	6.41	5.93	6.52

The best codelength ratio was order 1. This again could be due to the context of dyslexia text and the errors made. Although the order 1 model performs best in terms of both compression and classification, the higher orders still produced effective classification performance.

5.3.3 Experiment to Compare PPM with Other Classifiers

The purpose of this final experiment was to determine how the PPM classification method compares to other methods such as Multinomial Naïve

Bayes and Support Vector Machines. The Multinomial Naïve Bayes and Support Vector Machines classifiers have been described in Chapter 2. In the experiment, the Weka machine learning toolkit (Hall et al., 2009) was used for these methods. To classify text documents using Weka, the data set was preprocessed using a *string-to-word-vector-filter* and a *CharacterN-Gram* tokeniser. The classification F_1 score results are shown below in Table 5.8.

Table 5.8: Summary of F_1 experimental results for dyslexia classification using cross-validation.

Algorithm	BDAC vs. ALC	BDAC vs. BNDAC
MNB	0.96	0.78
SVM	0.98	0.81
PPM	0.99	0.71
MNB with bi-graph replacement	—	0.96
SVM with bi-graph replacement	—	0.93
PPM with bi-graph replacement	—	0.99

The algorithms provided similar results in distinguishing between the BDAC and ALC text. With regard to distinguishing between the BDAC and BNDAC texts, PPM without bi-graph pre-processing produces a lower F_1 score compared to MNB and SVM. However after pre-processing, the F_1 score increases to 0.99 outperforming the other two algorithms.

5.4 Conclusion

This chapter investigated a new form of text classification in order to test whether it is possible to distinguish between text written by people who are dyslexic with text written by people who are not dyslexic. The chapter specifically investigated using a text compression based classification method using the PPM algorithm with different orders to classify the dyslexic text.

Different experiments were conducted. An initial experiment using non-dyslexic text from an Arabic learner corpus (ALC) achieved very high accuracy using an order 1 PPM model. One possible explanation is that the dyslexic and non-dyslexic text used in the experiment were relatively easy to distinguish because they comprised different styles and were collected for different types of people in different ways. This led to the creation of a new non-dyslexic Arabic corpus (BNDAC) which used the same style and age as the dyslexia Arabic corpus (BDAC). Results showed that a bi-graph pre-processing method combined with PPM achieved the best classification results for the Arabic texts. Moreover, two other algorithms, SVM and MNB, were used to compare the results obtained with PPM. The experimental results show that using PPM to identify dyslexic text yielded the best performance.

Further sets of experiments investigated another language (English) for dyslexia classification to confirm if the result was applicable beyond the Arabic. The preliminary experiments produced similar results at distinguishing between the dyslexic and non-dyslexic texts. However, further experiments are required and these go beyond the scope of this thesis which is focussed on dyslexic texts in Arabic.

Chapter 6

Automatic Correction of Arabic Dyslexic Text

6.1 Introduction

Spelling errors have a significant influence on the way a person is perceived within a community and the frequency of such errors are often viewed as being linked to a person's intelligence (Rello et al., 2015). In this context, Graham et al.'s finding is that technologies (e.g. spellcheckers) can be used to aid people with dyslexia in order to minimise the incidence of spelling errors in their writing (Graham et al., 2001). Furthermore, as noted by Hiscox et al. (2014), automatic spelling correction can increase the motivation of people with dyslexia to write, thus elevating both their quality of life and the quality of their writing.

Therefore, this chapter describes a new system called Sahah “صحح” [E: “Correct” B: “SHH”] for the automatic spelling correction of dyslexic Arabic text.

The system described in this chapter uses the PPM compression-based language model and edit operations to generate possible alternatives for each error. The correct alternative for each error word is then selected automatically using the compression codelength. This chapter empirically shows how dyslexic errors in Arabic text can be corrected.

This chapter is organised as follows. Section 6.2 discusses spelling correction functions. Section 6.3 describes the Sahah system. After that, Section 6.4 discusses the evaluation methodology and Section 6.4.2 presents the experiments that were used to evaluate the effectiveness of the Sahah system. Firstly, the accuracy of the system is evaluated using an Arabic corpus (the BDAC) containing errors made by people with dyslexia. Secondly, the results of the system are compared with the results obtained using word processing software and the Farasa¹ tool. Section 6.5 concludes the chapter.

This chapter has been published in the Computers Journal (Alamri and Teahan, 2019).

6.2 Spelling Correction Functions

There are two functions that commonly appear in spelling correction tools namely a automatic correction function and a spellchecking function. These may seem similar, but they work differently. A spellchecker flags uncorrected words in the document and provides potential alternatives, called a suggestion list. In contrast, the purpose of the automatic correction function is to correct spelling errors automatically in the text without the need

¹<http://qatsdemo.cloudapp.net/farasa/>

to manually choose the word from a suggestion list. This is also called “autocorrect” and “text replacement” (Liensberger, 2015).

Sean Douglas, an internet broadcaster who is dyslexic, highlighted some of the issues relating to spelling correction for a person who is dyslexic: *“I generally have two options to deal with spelling mistakes; stop my writing and address every red line as I make a mistake, or wait till I get to the end to go through each spelling mistake one by one. While the built-in spell check in programmes like MS Word are pretty comprehensive, the extra time and fatigue caused by using them is far from desirable.”*

Spellcheckers can help users to self-monitor typos; they can also help users that have the cognitive ability to choose the correct spelling from a suggestion list (Berninger et al., 2008; Berninger and Wolf, 2016). MacArthur et al. (1996) and Montgomery et al. (2001) found that the spellchecker is most effective if the correct spelling is provided in the top three of the suggestion list. Users with dyslexia may face difficulty in choosing the correct word out of the ones suggested, so it is advisable to keep the suggestion list as short as possible (Leahy, 2002). As stated in the Douglas quote above, with spellcheckers, the writer needs to make an extra effort to correct errors. Therefore, the researcher of this thesis believes that an effective spelling correction tool for writers with dyslexia would be one that corrects the text automatically without requiring that the writer chooses the right word out of the suggested list.

6.3 The Sahah System for the Automatic Spelling Correction of Dyslexic Arabic Text

In order to propose an efficient spelling correction for dyslexic Arabic text, it is necessary to study and categorize the error patterns of dyslexia which has been done in Chapter 4. As mentioned earlier, the Sahah system is intended to correct dyslexic text automatically by using both a language model based on the PPM text compression scheme in addition to edit operations.

The workflow of the proposed Sahah system (see Figure 6.1) starts with transliteration from the original Arabic input text using the Buckwalter transformation in order to deal with Arabic text more effectively as mentioned in Chapter 2, Section 2.7.1.4, and consists of three stages. The first stage (Stage 1) is a pre-processing stage. The second stage (Stage 2) is a detection and correction stage that contains three further sub-stages: a sub-stage for error detection using a dictionary and two sub-stages for correction. The first sub-stage (2a) uses a PPM model to correct dyslexic errors according to their context. The second sub-stage (2b) is for error detection and uses an AraComLex dictionary (Attia et al., 2012). The third sub-stage (2c) is based on edit operations to generate a candidate list, then the codelength of the surrounding trigram is calculated in order to score and choose one word from the candidate list for the error.

The final stage (Stage 3) is the post-processing stage. The Sahah system ends with the reverse of the Buckwalter transliteration back to Arabic text. More details about each stage are described below.

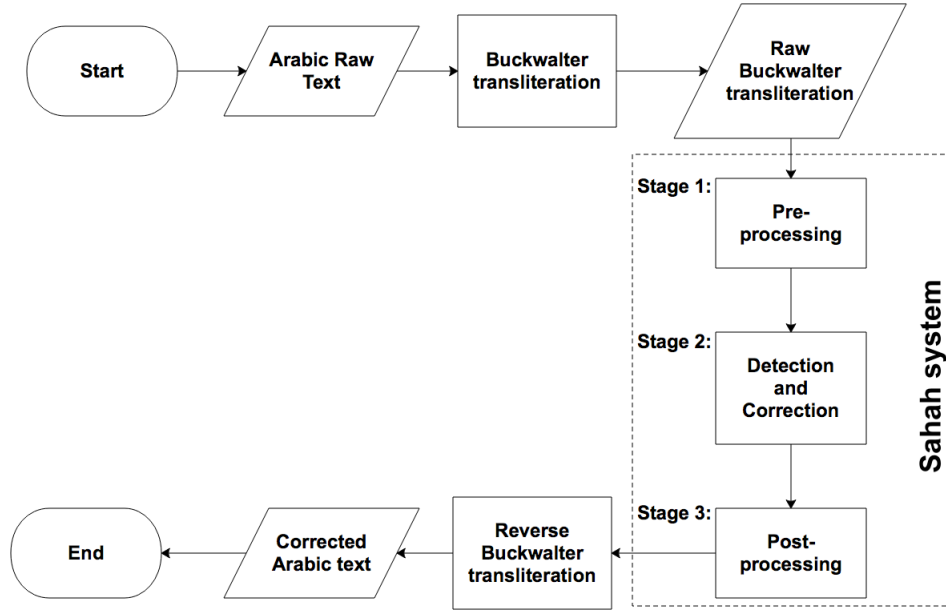


Figure 6.1: Workflow of the Sahah system.

6.3.1 Pre-processing Stage

While preparing the data for the process of error detection and correction, some errors such as tackling of split words and repeated characters were identified as causing ‘noise’ within the data, complicating the process. The analysis of dyslexic texts revealed that people with dyslexia sometimes divide Arabic words into two. This division could be due to the word having a short vowel or due to the pronunciation of the word. However, in some cases, a person with dyslexia inserts a space after prefixes or before suffixes; for instance, “أناها” [B: “<n hA”] to represent the word “إنها” [E: “it” B: “<nhA” R: “annaha”], which is not acceptable in Arabic texts. The characters “ها” [B: “hA”] are an Arabic suffix; thus, the way to cover the space insertion was inspired by a light stemming process, which refers to a process of removing prefixes and/or suffixes, without recognising patterns, dealing with infixes or finding roots (Larkey et al., 2002). However, instead of removing the prefix

or suffix, we concatenated the two together as part of the pre-processing process; for example, as in the word “أن” [B: “<n”] and the suffix “ها” [B: “hA”] to be “أنها” [E: “it” B: “<nhA” R: “annaha”].

The most common prefixes and suffixes based on the dyslexia corpus analysis were selected as shown in Table 6.1.

Table 6.1: The most common prefixes and suffixes based on the dyslexia corpus analysis.

Prefixes	Arabic	Buckwalter	Suffixes	Arabic	Buckwalter
	لل	ll		تما	tmA
	ف	f		ها	hA
	ك	k		وا	wA
	ل	l		نا	nA
	ا	A		تا	tA
	كا	kA		تي	ty
	ال	Al		ن	n
	با	bA		ه	h
	إل	<l		ت	t
	فا	fA		ي	y
	بي	by		ة	p

Additionally, the pre-processing stage covered the case where ‘p’ [B: ‘p’] is used in the middle of the word, as the character ‘p’ [B: ‘p’] only appears in the last position in words. Consequently, it is replaced with the character ‘t’ [B: ‘t’], as it is the most likely intended character. For example, the erroneous word “مكة” [B: ‘mkpbp’] is replaced by “مكتبة” [B: ‘mktbp’].

Hassan et al. (2014) removed the incorrect redundant characters from the

word. Likewise, the pre-processing stage in the Sahah system corrected the redundant characters, but with some modification. The modification is that in the Arabic language, there are some words in which a character can be repeated twice; for instance, “ممتاز” [E: “Excellent” B: “mmtaz”] repeats the character ‘م’ [B: ‘m’] twice. Therefore, characters that were repeated more than twice were reduced to just two repeated characters because no Arabic word contains three consecutive characters. However, there are some characters that can not be repeated consecutively twice, which are: ا, ل, ء, ع, ي and ة. This case is solved by reducing the repeated characters to one. Table 6.2 illustrates the way repeated characters were removed.

Table 6.2: Cases of removing the redundant characters.

Error	Intended word	After pre-processing
المملك [B: “Almmmlk”]	الملك [E: “The king” B: “Almlk”]	المملك [B: “Almmk”]
الصورة [B: “AllSwrh”]	الصورة [E: “The picture” B: “AlSwrp”]	الصورة [B: “AllSwrh”]
سمايه [B: “smAAyh”]	سمائه [E: “His sky” B: “smA}h”]	سمايه [B: “smAyh”]

It was found that if the pre-processing step was introduced prior to the error detection and correction stage, it would resolve the issue of the split words and repeated characters, which meant that the accuracy of the detection and correction stage would be enhanced. For example, in Table 6.2 there is the word “سمايه” [B: “smAAyh”], which contains the redundant characters “اا” [B: “AA”]. If the Sahah system does not include the pre-processing stage, Sahah will change the word “سمايه” [B: “smAAyh”] to “سماوية” [B: “smAwyp”], which is not the intended word. However, by using the pre-processing stage, the Sahah system can correct the error to the intended word, which is “سمائه” [E: “His sky” B: “smA}h” R: “smi’ah”]. Therefore, the pre-processing stage described above included the tackling of split words and repeated characters.

6.3.2 Error Detection and Correction Stage

This stage was as stated divided into three sub-stages: sub-stage (2a) that employed the PPM compression-based language model; sub-stage (2b) that employed error detection based on a dictionary; and sub-stage (2c) that employed edit operations to generate the candidate list, then score the candidate list based on the codelength. The workflow of stage 2 is shown in Figure 6.2; also more detail about each sub-stage are explained below.

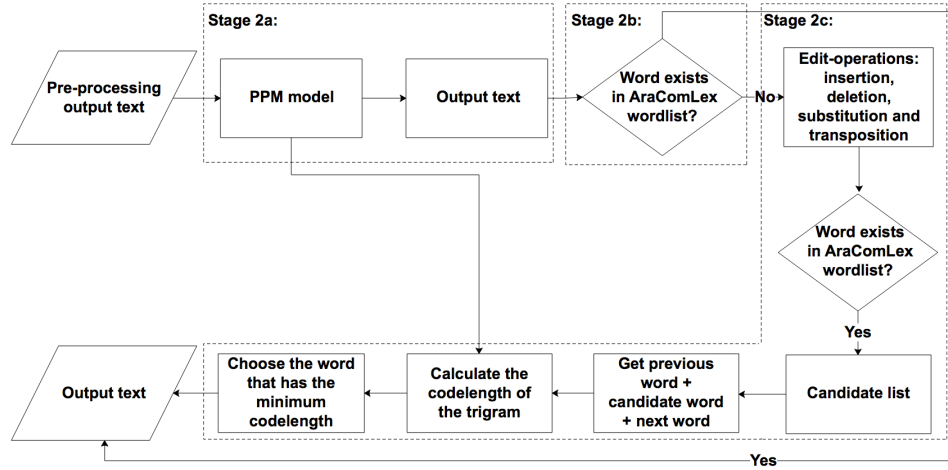


Figure 6.2: Workflow of the error detection and correction stage of the Sahah system.

According to Abu-Rabia and Sammour (2013) one of the main spelling rules in Arabic which students find it difficult to master concerns the writing of Hamza. Thus, this order of sub-stages arose after discovering the importance of correcting some errors first such as the Hamza error type before checking the word with the dictionary and generating the candidate list.

For example, the sentence “وتبين انظمه التشغيل للحاسوب” [B: “wtbyn AnZmh Alt\$ygl llHAswb”] contains three errors types. The second word “انظمه” [B: “AnZmh”] contains two types of errors; ‘ا’ [B: ‘A’] needs to be replaced with ‘ي’ [B: ‘<’] and

‘:’[B: ‘p’] needs to be used instead of ‘.’[B: ‘h’]. The third word is “التشغيل” [B: “Alt\$ygl”] with the transposition of “غيد” [B: “gy”] to “يدغ” [B: “yg”].

Table 6.3 illustrates the result of using the PPM correction sub-stage (2a) first, then the edit operation correction sub-stage (2c) next, and vice versa.

Table 6.3: Example of an error that corrects with a different ordering of sub-stages.

	PPM correction	Detection → Edit operation
Order 1	وتبين أنظمة التشغيل للحاسوب [B: “wtbyn <nZmp Alt\$ygl ll- HAswb”]	وتبين أنظمة التشغيل للحاسوب [B: “wtbyn <nZmp Alt\$gyl ll- HAswb”]
	Detection → Edit operation	PPM correction
Order 2	وتبين نظمه التشغيل للحاسوب [B: “wtbyn nZmh Alt\$gyl ll- HAswb”]	وتبين نظمه التشغيل للحاسوب [B: “wtbyn nZmh Alt\$ygl ll- HAswb”]

As shown in Table 6.3, if the PPM correction sub-stage (2a) is utilised first then followed by the detection sub-stage (2b) and edit operation sub-stage (2c), the system can correct all the errors in the sentence. Contrariwise, if we utilised the detection (2b) and edit operation sub-stage (2c) first then the PPM correction sub-stage (2a) next, the system can correct the transposition error only.

6.3.2.1 Sub-stage 2a: PPM Correction

This sub-stage is based on using the PPM language model (PPM has been described in detail in Chapter 2, Section 2.7.1). For the correction process, we use an encoding-based noiseless channel model approach as opposed to the decoding-based noisy channel model (Teahan, 2018). As Teahan (2018) mentioned, instead of performing a decoding of the observed message we

can perform a search to find the best encoding of the target message. We use ‘observed→corrected’ rules, which denotes the transformation from the observed state to the corrected state when the noiseless channel correction process is applied. The PPM model was applied in order to correct the errors for a given set of transformations rules by using a Viterbi-based algorithm to search through all possible alternative spellings for each character in order to find the most compressible sequence from these possible alternatives at the character level.

The Viterbi algorithm guarantees that the alternative with the best compression will be found by using a trellis-based search: all possible alternative search paths are extended at the same time, and the poorer performing alternatives that lead to the same conditioning context are discarded (Teahan, 1998).

As previously mentioned in Chapter 2, Section 2.7.1, there are two mechanisms of PPM; update exclusions (UE) and without update exclusions (WUE). In order to check the performance of the model with and without update exclusions, two experiments were conducted.

To perform the experiment, ten percent of the Bangor Dyslexia Arabic Corpus (BDAC) was used. Firstly, two models were created from standard Arabic text as represented in a training corpus; one model with update exclusions and one without update exclusions. The training corpus consisted of the BACC corpus, a 31,000,000-word corpus called the Bangor Arabic Compression Corpus (BACC) created by Alhawiti (2014) for standardising compression experiments on Arabic text. Alkahtani (2015) developed a parallel corpus that includes 27,775,663 words in Arabic, based on corpora from Al Hayat articles and the open-source online corpora database and from the

King Saud University Corpus of Classical Arabic (KSUCCA), which is part of research attempting to study the meanings of words used in the Holy Quran through analysis of their distributional semantics in contemporaneous texts (Alrabiah et al., 2013). The above three corpora combined are jointly referred to here as the BSK corpus. A large text corpus was needed in order to develop a well-estimated language model. This need was met by the BSK corpus.

Then, these models were used in the initial sub-stage (2a). Table 6.4 shows the findings indicated that using update exclusions and without update exclusions.

Table 6.4: Improving the model of sub-stage 2a.

Sub-stage	Detection				Correction			
	Rec.	Prec.	F ₁ score	Acc.	Rec.	Prec.	F ₁ score	Acc.
2a								
UE	0.61	0.91	0.73	0.82	0.22	0.79	0.34	0.66
WUE	0.52	0.92	0.67	0.79	0.26	0.86	0.40	0.69

As a result of the above experiments, the model without update exclusions was selected for sub-stage (2a). Subsequently, two models with and without update exclusions were created using the BSK to see which one worked better in calculating the codelength (Sub-stage 2c). The results are presented below in Table 6.5:

Table 6.5: Improving the model of sub-stage 2c.

Sub-stage	Detection				Correction			
	Rec.	Prec.	F ₁ score	Acc.	Rec.	Prec.	F ₁ score	Acc.
2c								
UE	0.75	0.93	0.83	0.88	0.40	0.89	0.55	0.74
WUE	0.75	0.93	0.83	0.88	0.41	0.89	0.56	0.74

The results of these different experiments revealed that the language model

without update exclusions produced improved results by approximately 2% over the model with update exclusions, which is compatible with the findings of Al-kazaz for cryptanalysis (Al-kazaz et al., 2018). Therefore, the variant of PPM model adopted in this chapter was without update exclusions (WUE).

For example, in order to correct the erroneous word “احمد” [B: “AHmd”], which contains one error, ‘ا’ [B: ‘A’] is replaced with ‘إ’ [B: ‘>’]. The correct version is “أحمد” [B: “>Hmd”]. The Viterbi-based search algorithm generated possible alternative for each character by using the set of transformations rules shown in Table 6.6.

Table 6.6: Set of transformations rules from the DECA used for sub-stage (2a) in the Sahah system.

ا → أ	ظ → ض
ا → إ	ض → ظ
ا → ي	ي → ا
أ → ء	ا → ن
ئ → ي	أ → ن
ؤ → و	أ → ن
وا → و	ن → ء
ت → ة	ن → ء
ة → ت	ن → ء
ه → ة	و → ء
ه → ة	ي → ء

From the example, the character ‘ا’ [B: ‘A’] can be (‘إ’ [B: ‘<’], or ‘إ’ [B: ‘>’], or ‘ي’ [B: ‘Y’]).

The Table 6.7 shows the output of utilising the PPM language model to calculate the codelengths for the corrections using the large BSK training

corpus. Thus, the smallest codelength was given to the word “أحمد” [B: “>Hmd” R: “aḥmd”], which is the correct version of the word.

Table 6.7: The codelength of possible alternatives spellings by using the confusions in Table 6.6 for the erroneous word “أحمد”.

Transformation	Codelength
‘ا’ [B: ‘A’] → ‘ا’ [B: ‘>’]	أحمد [B: “>Hmd”] = 12.26 bits
‘ا’ [B: ‘A’] → ‘ا’ [B: ‘<’]	إحمد [B: “<Hmd”] = 24.94 bits
‘ا’ [B: ‘A’] → ‘ي’ [B: ‘Y’]	يحمد [B: “YHmd”] = 28.06 bits
‘ا’ [B: ‘A’] → ‘ا’ [B: ‘A’]	احمد [B: “AHmd”] = 15.95 bits

The pre-processing stage and the PPM correction stage covered many error types from the DECA, which include the Hamza, Confusions, Diacritics and Form, but it did not include the common errors, which are omission, addition, substitution and transposition. Therefore, Norvig’s approach (Norvig, 2009) was deemed appropriate for these type of errors. However, first it is necessary to know whether or not the word is an erroneous word; hence, Sub-stage 2b is required.

6.3.2.2 Sub-stage 2b: Error Detection

The most direct means of detecting error words is to search for each word in a dictionary and report the words that are not located therein. Based on this principle, an open-source dictionary was used to detect errors with a list containing nine million Arabic words. The words in this dictionary were generated automatically from the AraComLex open-source finite state transducer (Attia et al., 2012), since it is a free resource that has proven to be effective in previous studies to either correct or detect spelling errors (Shaaan et al., 2012; Hassan et al., 2014; Zaghouani et al., 2015).

Prior to checking whether a word is in the AraComLex or not, any diacritical marks have to be removed for two reasons. The first reason is that the dyslexic corpus itself does not contain diacritical marks. People with dyslexia have diacritical issues – for example, they write the diacritical Tanwin as character ‘ن’ [B: ‘n’], but do not usually put diacritical marks in their writing. The second reason is that the AraComLex does not contain diacritical marks. If the input word was not located in the AraComLex dictionary as illustrated in Figure 6.2, it was considered to contain a spelling error and was passed to the edit operation Sub-stage 2c.

6.3.2.3 Sub-stage 2c: Edit Operations

This sub-stage is based on using edit operations, which consist of applying addition (add a letter), omission (remove letter), substitution (change one letter to another) or transposition (swap two adjacent letters) of the error word, and returns a set of all of the edited words that can be achieved using one or two edit operations. A set of candidate corrections is then generated, including real and non-real words. The candidate list was filtered with reference to an open-source dictionary (AraComLex), commencing with the list of known words for the first edit operation, if any existed, and proceeding to the list of known words for the second edit operation.

Once the Sahah system has generated the candidate list, the PPM language model is run to calculate the codelength of the candidate surrounding trigram (previous word, candidate word and next word), then returns the candidate word with the lowest candidate trigram codelength. Using the previous example in Section 6.3.2 above, “وتبين انظمه التشغيل للحاسوب” [B: “wtbyn AnZmh Alt\$ygl llHAswb”], following sub-stage 2a, which corrected the

second word “وتبين أنظمة التشغيل للحاسوب” [B: “wtbyn >nZmp Alt\$ygl llHAswb”], there was still an error in the third word “التشغيل” [B: “Alt\$ygl”], which was under the common category. Table 6.8 shows the candidate list for the error word “التشغيل” [B: “Alt\$ygl”]:

Table 6.8: Codelengths for different candidate trigrams for a sample correction.

Candidate word	Candidate trigram	Codelength (bits)
“التشغيل” [B: “Alt\$ygl”]	“أنظمة التشغيل للحاسوب” [B: “>nZmp Alt\$ygl llHAswb”]	100.82
“التشاكل” [B: “Alt\$Agl”]	“أنظمة التشاكل للحاسوب” [B: “>nZmp Alt\$Agl llHAswb”]	106.45

The lowest codelength is for the candidate word “التشغيل” [B: “Alt\$ygl”], which required 100.82 bits to encode the trigram. Therefore, the Sahah system corrected all errors in the following sentence: “وتبين أنظمة التشغيل للحاسوب” [E: “and show the operating systems of the computer” B: “wtbyn >nZmp Alt\$ygl llHAswb” R: “w tbyn anẓmat altšğyl llḥaswb”].

6.3.3 Post-processing Stage

The space omission problem was tackled using word segmentation during the post-processing stage. Word segmentation is the process of determining the most compressible sequence when all possible insertions of space are considered. It is an important task for some natural language processing applications, such as speech recognition. Character-based PPM models with the use of the Viterbi algorithm (Viterbi, 1967) has achieved a high accuracy rate for the word segmentation of English and Arabic text (Teahan, 1998; Alhawiti, 2014).

In order to correct the segmentation of dyslexic errors where spaces had been

omitted, the order five character-based PPM model was first trained on the three corpora (BSK). Two segmentations are possible for each character: the character itself and the character followed by a space. In order to find the most probable segmentation sequence that exhibits the best encoding performance, as determined by the PPM language model, the Viterbi-based search algorithm via the noiseless channel model approach was used again to find the best segmentation as measured by the sequence of text with spaces inserted that had the lowest compression codelength. For example, a sample incorrect sequence is “الطائر غرد” [B: “AlTA}rgrd”], while the intended sequence is “الطائر غرد” [E: “the bird is chirping” B: “AlTA}r grd”].

The last step in the Sahah system is the reverse transliteration of the output back into Arabic.

6.4 Evaluation

This section discusses the evaluation methodology and the experiments that were conducted to evaluate the performance of the Sahah dyslexic Arabic spelling correction system that is presented in this chapter.

6.4.1 Evaluation Methodology

There are five possible outcomes of the Sahah system. These cases are based on those proposed by Pedler (2007). However, the case where “the error was considered by the program but wrongly accepted as correct” was not applicable in the Sahah system, so it was not adopted. This is because once the Sahah system detected the error, it is either changed to a correct word or an incorrect alternative word. Errors can be dealt with in the first

three cases below, and correctly spelt words can be dealt with in the last two cases as below:

Corrected case: The error is detected and replaced with the intended word (**Case I**).

Incorrect alternative case: The error is detected and replaced with an incorrect alternative (**Case II**).

Missed case: The error is not detected, and therefore, the system does not correct it (**Case III**).

Skipped case: The word that is spelt correctly is accepted (**Case IV**).

False alarm case: The word that is spelt correctly is changed (**Case V**).

The sentence below illustrates the five possible outcomes as represented by the error \rightarrow correction form:

The raw text: “Thei were not the onle ones leving on that land.”

The gold-standard text: “They were not the only ones living on that land.”

Case I: Thei \rightarrow They.

Case II: leving \rightarrow leaving.

Case III: onle \rightarrow onle.

Case IV: were \rightarrow were.

Case V: land \rightarrow island.

The evaluation methodology used in this study is based on recall, precision, F_1 score and accuracy, which are common natural language processing measures. The gold-standard correction for each spelling error was manually prepared as described in Chapter 4.

The two main functions of the Sahah system are error detection and error correction. The evaluation of the Sahah system was therefore separated into two parts: error detection evaluation and error correction evaluation.

Error detection evaluation: The error detection function evaluates whether a word is detected when compared with the gold-standard manual annotation. Recall, precision, F_1 score and accuracy are calculated using equation 2.1, 2.2, 2.3 and 2.4 in Chapter 2 as follows:

The total number of corrected words and incorrect alternative words (Case I and Case II) gives the TP , while the FN is the number of missed words (Case III), FP is the number of false alarm words (Case V) and TN is the number of skipped words (Case IV).

Error correction evaluation: The error correction evaluation is calculated by determining whether a word has been successfully corrected based on the gold-standard manual annotation. Recall, precision, F_1 score and accuracy can then also be calculated using equation 2.1, 2.2, 2.3 and 2.4 in Chapter 2 as follows:

TP is the number of corrected words (Case I) and TN is the number of skipped words (Case IV), while FN is the total number of incorrect alternative words and missed words (Cases II and III) and FP is the number of false alarm words (Case V).

The difference between the two confusion matrices is only in Case II, moving from TP in detection to FN in correction. The reason is to consider the corrected cases only in order to evaluate the correction.

6.4.2 Experimental Results

The Sahah system developed for this study was evaluated in two ways: (i) using the BDAC corpus that consisted of text written by people with dyslexia; and (ii) using a comparison with commonly-used spellcheckers/tools.

(i) Experiment using the BDAC corpus: This experiment used the BDAC corpus (28,203 words). The recall rate, precision and F_1 score for the pre-processing stage and Sub-stage 2a using the PPM language model are presented in Table 6.9.

Table 6.9: Detection and correction results after the pre-processing stage and sub-stage 2a of the Sahah system.

	Rec.	Prec.	F_1 score	Acc.
Detection	0.53	0.93	0.68	0.84
Correction	0.28	0.88	0.43	0.76

When all stages and sub-stages are taken into consideration, the Sahah system achieved a better result as shown in Table 6.10.

Table 6.10: Detection and correction result after all stages and sub-stages of the Sahah system were applied.

	Rec.	Prec.	F_1 score	Acc.
Detection	0.75	0.93	0.83	0.90
Correction	0.43	0.89	0.58	0.80

The F_1 score for correction increased by 15% when the edit operations Sub-stage 2c was used. It is clear that the inclusion of Sub-stages 2b and 2c led to a higher rate of recall, precision, F_1 score and accuracy. Some examples of Sahah output using BDAC are shown in Appendix 8.

(ii) Experiment using a comparison with commonly used spellcheck-

ers/tools: For the experimental comparison with commonly-used tools, there are two parts: namely detection comparison and correction comparison.

Detection Comparison

For our comparison, we compared the results of the Sahah system against Microsoft Office and Ayaspell 3.0 used in OpenOffice because it is a widely-used word processing software. Furthermore, there are a number of previous studies that used Microsoft Office and Ayaspell to evaluate their approach (Noaman et al., 2016; Attia et al., 2012; Mars, 2016; Rello et al., 2015). The results in Table 6.11 list recall, precision, F_1 score and accuracy by using the BDAC corpus.

Table 6.11: Detection comparison using the Bangor Dyslexia Arabic Corpus (BDAC) corpus.

Spellchecker tool	Rec.	Prec.	F_1 score	Acc.
MS word	0.47	0.97	0.63	0.83
Open Office Ayaspell	0.52	0.98	0.68	0.85
Sahah	0.75	0.93	0.83	0.90

The assessment of our system’s ability to detect errors is based on the F_1 score. Sahah’s 0.83 (shown in bold font) was significantly higher than that for both Ayaspell for OpenOffice (0.68) and Microsoft Word (0.63).

Correction Comparison

The Sahah system does not show a suggestion list, which means there is no need for human interaction to replace erroneous words. Thus, the spellcheckers investigated above in Table 6.11 are not compatible with our correction system that was investigated for these experiments. Therefore, for comparison purposes, the results obtained from this study for the Sahah system in

Section 6.4.2 above were compared to the results obtained using the Farasa tool, which is a text processing toolkit for Arabic text.

Farasa comprises a segmentation/tokenisation module, a part-of-speech tagger, an Arabic text diacritizer and spellchecker. Farasa is available online and operates in a similar way to the Sahah system in this study. The Farasa tool corrects the text automatically without showing a suggestion list. The use of Farasa has been described in two papers (Mubarak and Darwish, 2014; Mubarak et al., 2015). Both studies produced results with respect to correcting Arabic news, native and non-native text.

The results in terms of recall, precision, F_1 score and accuracy using the BDAC corpus are presented in Table 6.12.

Table 6.12: Correction comparing the Sahah system with the Farasa tool.

Tool	Rec.	Prec.	F_1 score	Acc.
Farasa	0.23	0.84	0.36	0.74
Sahah	0.43	0.89	0.58	0.80

When compared with the Farasa tool, the Sahah system achieved a higher F_1 score.

Although the Sahah system produced good recall, precision and F_1 score rates as discussed above, it could not detect some errors (Type I) or could not correct some errors that were detected (Type II). The errors can be categorised as follows:

- Type I: The Sahah system in some cases could not detect an error if the word used matched with a word in the dictionary. Furthermore, it could not detect errors falling under the word boundary error category, for example the use of “لي عقولهم” [B: “ly Eqwlhm”] instead of “لعقولهم” [E:

“To their minds” B: “lyEqwllhm”] where both words are valid. However, it is worth noting that none of the widely-used word processing software, Microsoft Office and Ayaspell 3.0 used in OpenOffice or the Farasa tool referred to above can detect this type of error.

- Type II: If more than one letter in the word is deleted or added, it makes the word hard to correct. In such cases, the Sahah system inserted an alternative word. For example, instead of the erroneous word “التر” [B: “Altr”], which is missing three letters, the Sahah system substituted it with “البر” [B: “Albr”] when the intended word was “التربية” [B: “Altrbyp”]. When the erroneous word contained more than three types of errors, the Sahah system could easily detect the error, but could not correct it, for example, “اليلاملاي” [B: “AlylAmlAy”] which was used instead of “الإملائية” [B: “Al<mlA}yp”]. This contained five errors that were detected by the Sahah system, which then exchanged it with the incorrect alternative “اللام لأي” [B: “Al}lAm l>y”].
- Type II: An incorrect alternative occurred when the wrong candidates were chosen on the basis of the codelength of the trigram. For example, for “الصوص” [E: “The thieves” B: “AlSwS”], the candidates’ list included “الصوص” [E: “The thieves” B: “AllSwS”] (94.72 bits) and “الصوت” [E: “The voice” B: “AlSwt”] (89.46 bits). The candidate list contained the intended word, but the smallest codelength was for [B: “AlSwt”], which is an incorrect alternative in this case.
- Type II: Addition words, deletion words or synonyms written for a word during dictation time such as “البيت” [E: “Home” B: “Albyet”] instead of “المنزل” [E: “House” B: “Almzel”] fall outside the scope of this study as they do not contain errors and are very rare in the

BDAC corpus.

6.5 Conclusion

This chapter introduced the Sahah system that automatically detects and corrects errors in Arabic text written by people with dyslexia. The Sahah system has three stages: a pre-processing stage, that corrects split words and repeated characters; the second stage that uses the character based PPM language model to identify the best correction for the erroneous words, and also uses edit operations (omission, addition, substitution and transposition) and the correct alternative for each error word is chosen on the basis of the compression codelength of the enclosing trigram; and the post-processing stage that addresses the spaces that had been omitted. It does this by using a character-based PPM method in order to correctly segment the errors caused by people with dyslexia.

The BDAC containing errors made by people with dyslexia was used to evaluate the performance of the Sahah system presented in this chapter. This system significantly outperforms the Microsoft Word and Ayaspell systems for the detection stage and the Farasa tool in the correction stage. The approach provided good results compared with the other tools, with an F_1 score of 0.83 for detection and an F_1 score of 0.58 for correction.

Chapter 7

Conclusion

7.1 Introduction

This chapter discusses the work conducted in this thesis, and the experiments performed. It also reviews the study's aim and objectives, and the answers to the research questions. Future work is also discussed.

7.2 Summary of the Thesis

This study investigated the feasibility of employing a technique based on text compression, and specifically investigated the effectiveness of PPM, for tackling the problem of classifying and correcting of Arabic dyslexic text. Several experiments addressing the classification problem were conducted, resulting in significant improvements to the accuracy of Arabic dyslexic text classification. In addition, a new system for the automatic spelling correction of dyslexic Arabic text was developed.

This research study commenced with a review of Chapter 2, which reviewed the dyslexia. It discussed fundamental characteristics of the Arabic language, highlighting and describing the spelling errors produced by people with dyslexia and Arabic writers with dyslexia and corpus linguistics. It explored the methods that can be used to classify and correct text. It also presented an extensive review of the PPM compression scheme, together with a description of how the codelength within this scheme is calculated.

Chapter 3 discussed the improvement and enlargement of the Bangor Dyslexia Arabic Corpus (BDAC), the present content of which was gathered from both male and female students in Riyadh, Saudi Arabia, who had been professionally diagnosed with dyslexia, and who were from a similar population, in terms of their age and education, and whose native language was Arabic. The chapter also described the ways in which dyslexia is identified in schools in Saudi Arabia, and explained the procedure employed for the collation of the BDAC, the text for which was collected from different sources: homework produced by people with dyslexia, text provided by the parents of children with dyslexia, and a form answered by people with dyslexia. In total, the current BDAC corpus consists of 28,203 words. As the literature review evidenced, the BDAC is the only dyslexia corpus for Arabic text.

Chapter 3 also discussed how the text in the BDAC was transcribed into an electronic format, explaining that the transcription was conducted manually by the researcher and one volunteer. It then discussed the subsequent analysis of the BDAC documents, which contained words, sentences, and paragraph texts, together with the analysis of the participants' information. The analysis determined the frequency of the words and characters in the BDAC, and found that the word with the most frequency was 'في' [B: 'fy']

and the character with the most frequency was ‘ا’ [B: ‘A’].

Following the analysis in Chapter 3, Chapter 4 discussed the development of a new dyslexic error classification scheme for Arabic texts (DECA), explaining that it was based on an analysis of previous studies of dyslexic errors, which together provided a platform for understanding and analysing the specific errors made by people with dyslexia. The resulting classification scheme was comprised of 37 types of errors, grouped into nine categories. The chapter also discussed the three evaluations that were conducted to assess the DECA’s reliability and effectiveness. The first evaluation determined whether the error tags were sufficiently clear, and assessed whether any types were absent from the DECA. Two annotators (A1 and A2) conducted the first evaluation. Both found that the types in the DECA were clear, and A1 suggested the addition of two further types: Repeated Letters; and Written Form in Beginning, Middle, or End. Consequently, Version 1 of the DECA was edited to include these two types. The second evaluation involved a questionnaire that was sent to two primary school teachers of children with dyslexia (T1 and T2). It was designed to assess whether the DECA included all of the errors produced by students with dyslexia, and whether the categories were appropriate. Both evaluators agreed that the table of the types of dyslexic errors was comprehensive. The final evaluation employed Kappa statistics to measure and compare the agreement between the annotators. It was conducted by A2 and T1 from the previous evaluations, together with the researcher of this thesis (N1). The agreement was 87% between T1 and N1, 88% between A2 and N1, and 84% between A2 and T1, thereby demonstrating a high degree of agreement between the annotators.

In addition, the process of the annotation of the BDAC was described. The

chapter concluded with an analysis of Arabic dyslexic errors, which revealed that some of the errors in the corpus occurred more than others, and that the highest number of errors for a specific category was for the Common errors category, followed by the Hamza category.

The two resources presented in Chapters 3 and 4, namely the BDAC and DECA, are extremely valuable, as they pave the way for an Arabic dyslexia corpus to be used for other purposes, such as for applications for Arabic people with dyslexia. Thus, Chapters 5 and 6 explored the use of the Arabic dyslexia corpus to classify and correct Arabic dyslexic text, which was the main aim of this thesis.

Chapter 5 investigated a new form of text classification, in order to determine whether it is possible to distinguish between text written by people with dyslexia, and text written by non-dyslexic people. The chapter specifically investigated the use of the PPM text compression scheme to classify the dyslexic text, using different orders (from 0 to 5). In terms of the dyslexia classification method employed, the dyslexia corpus and the non-dyslexia corpus were selected as representative of dyslexic text and non-dyslexia text, in order to train the PPM character-based language models. Two models, the dyslexia model and the non-dyslexia model, were used to compute the codelength ratio by compressing the test file, based on the training of the two models. The chapter described the three corpora, that were employed to create the dyslexic and non-dyslexia language models. Experiments were conducted in order to evaluate the quality of the PPM classification method for dyslexia classification. The first experiment used the BDAC and the Arabic Learner Corpus (ALC) corpus with the result showing that PPMD order 1 achieved an F_1 score of 0.99. The second experiment used a new non-dyslexic Arabic corpus (the BNDAC) that was built for this study,

that employed the same style as the BDAC, to determine the validity of the preliminary results of the first experiment. The initial results using the BDAC and BNDAC were not as expected, since the F_1 score of order 1 was 0.67. Consequently, a bi-graph replacement method was used, which significantly improved the compression performance. The bi-graph pre-processing method, combined with PPM, achieved the best classification results for this experiment, with an F_1 score of order 1 was 0.99.

The chapter concluded with a further experiment comparing PPM with SVM and MNB classification algorithms. The algorithms demonstrated an excellent result in distinguishing between the ALC and BDAC text, where the F_1 score for SVM and MNB were 0.98 and 0.96 respectively. The F_1 score were 0.81 and 0.78 for SVM and MNB algorithms produced a better result than the standard PPM, without bi-graph pre-processing, but after pre-processing, the F_1 score for PPM increased to 0.99, thereby outperforming the other two algorithms in distinguishing between the BNDAC and BDAC text. Overall, the results demonstrated that using PPM to identify dyslexic text yielded the best performance.

Chapter 6 addressed the problem of the automatic correction of spelling errors in Arabic text written by people with dyslexia, and demonstrated empirically how dyslexic errors in Arabic text can be corrected. It introduced the Sahah system, which used the PPM compression-based language model and edit operations. The Sahah system consisted of three stages. The first stage was a pre-processing stage that corrected split words and repeated characters. The second stage was a detection and correction stage that contained three further sub-stages. The first sub-stage (2a) used the PPM model to correct Arabic dyslexic errors, according to their context. The PPM model employed a Viterbi-based algorithm to search all possible

alternative spellings, in order to locate the most compressible sequence. The second sub-stage (2b) employed an AraComLex dictionary to detect errors; if the word was not in the dictionary, it was passed to the third sub-stage (2c), which used edit operations that generated a candidate list, then the codelength of the surrounding trigram was calculated to score the candidate, and select the most appropriate correction for the erroneous word. Finally, the post-processing stage addressed the spaces that had been omitted, using a character-based PPM method to correctly segment the errors produced by people with dyslexia.

In addition, Chapter 6 described the five possible outcomes of the Sahah system, and discussed the evaluation methodology, and how it evaluated the error detections and error corrections. The chapter discussed two experiments that were conducted to evaluate the performance of the Sahah. The first used the BDAC corpus, and second compared the new system with the spellchecker tools of Microsoft Word, Ayaspell systems, and the Farasa tool. The intermediate results of the first experiment, after employing the pre-processing stage, and sub-stage 2a of the Sahah system, revealed that the F_1 score for detection was 0.68, and the F_1 score for correction was 0.43. After all of the stages and sub-stages were taken into consideration, the Sahah system achieved a better result, with an F_1 score for detection of 0.83, and an F_1 score for correction of 0.58. The second experiment compared between the spellchecker tools of Microsoft Word, Ayaspell, and Farasa, and the results revealed that the Sahah system achieved a significantly higher F_1 score than that for Ayaspell (0.68), and Microsoft Word (0.63), while for correction, the Sahah system achieved a higher F_1 score than Farasa (0.36).

7.3 Review of Research Questions

The research questions designed for this study, which are listed in Section 1.2, were all addressed. The project demonstrated success in employing the PPM compression method for classifying and correcting Arabic dyslexic text. The PPM compression scheme performed well in different language modelling tasks, and was also successfully applied to classifying and correcting Arabic dyslexic text.

The specific research questions detailed in Section 1.2 were addressed as follows:

1. *What is an effective spelling error classification scheme for annotating and analysing Arabic dyslexic corpora?*

The DECA was comprised of 37 types of errors, grouped into nine categories, and was demonstrated to be effective for annotating the BDAC, and for analysing dyslexic errors. It was demonstrated in Chapter 4 that the DECA is clear, comprehensive, and effective, with a high degree of agreement of over 0.80 when used for annotating the BDAC.

2. *How well does a compression-based language modelling method, such as the Prediction by Partial Matching (PPM) text compression method, compare to two well performed algorithms such as Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM) for classifying a text that has been written by a person with dyslexia?*

As discussed in Chapter 5, which compared PPM results with other classification methods, namely Support Vector Machines and Multinomial Naïve Bayes, PPM produced an excellent F_1 score of 0.99 in

distinguishing between the ALC and BDAC text. With regards to distinguishing between the BDAC and BNDAC texts, PPM produced a lower F_1 score of 0.71, compared with Multinomial Naïve Bayes (0.78), and Support Vector Machines (0.81), using standard PPM without bi-graph pre-processing, but after pre-processing was applied, the F_1 score for PPM increased to 0.99, outperforming the other two algorithms. The PPM classification method was therefore deemed to be more effective than the other two algorithms.

3. *Can PPM, in conjunction with other methods, be effectively applied to correcting a text that has been written by a person with dyslexia?*

The experiments reported in Chapter 6 confirmed that the Sahah system, containing three stages namely pre-processing, detection and correction, and post-processing, was very effective for correcting Arabic dyslexic text, compared with other spellchecker tools, namely Microsoft Word, Ayaspell, and Farasa.

7.4 Review of Aim and Objectives

The aim and objectives of this study, as described in Section 1.3, were all successfully achieved. This thesis aimed to investigate the effectiveness of using the PPM compression method to classify and correct Arabic dyslexic text. Consistent with the above research questions, this project achieved the objectives as follows:

- *Review the extant literature regarding dyslexia, Arabic language, dyslexia spelling errors, corpus linguistics, text classification, spelling correction, and text compression.*

This objective was achieved in Chapter 2, in which the dyslexic Arabic language spelling errors made by people with dyslexia, and the related corpora were evaluated. In addition, this chapter also discussed the different methods of classifying and correcting text, and presented PPM, detailing how it functions.

- *Improve the existing Arabic corpus of texts written by people with dyslexia (the Bangor Dyslexia Arabic Corpus (BDAC))*

This improvement was achieved by the development of the Arabic dyslexia corpus (the BDAC) of 28,203 words, written by both male and female with dyslexia, aged between 8 and 13 years, which was discussed in Chapter 3.

- *Create a new dyslexic error classification scheme for Arabic dyslexic texts (DECA).*

This objective was achieved by the development of new dyslexic error classification scheme for Arabic (the DECA) comprised of 37 types of errors, grouped into nine categories, as discussed in Chapter 4. This scheme assists in analysing and annotating errors produced by people with dyslexia.

- *Develop and evaluate a method to classify whether or not a text has been written by a person with dyslexia, using the PPM compression scheme, and compare the performance of the PPM with other classification methods, such as the Multinomial Naïve Bayes (MNB) and Support Vector Machines (SVM), when they are employed for the purpose of classifying dyslexic text.*

This objective was achieved, as demonstrated in Chapter 5, which discussed the results showing that the PPM compression method is

effective for classifying dyslexic text, and that it outperforms other classification methods, such as the SVM, and the MNB.

- *Design and evaluate an automatic spelling correction system for correcting spelling errors in Arabic texts, produced by people with dyslexia, by comparing them with other spelling correction tools.*

This objective was achieved, as discussed in Chapter 6, by developing and testing a new system, called Sahah, that automatically corrects Arabic dyslexic text, using different stages and that it significantly outperforms other tools, like Microsoft Word, Ayaspell, and Farasa.

7.5 Future Work

Based on this research, the following are recommended as areas of further investigation:

- While the corpus provided insights into the writing of dyslexic Arabic people, and is appropriate for assisting writers with dyslexia, it can also serve as a platform for other researchers to build upon, as it can be employed as the first step to adding more text collected from adults with dyslexia in Saudi Arabia or other Arab countries.
- As a direct consequence, adding text to the corpus may enable the determination of emerging patterns of errors found in the writing of people with dyslexia in the Arab context, which in turn will assist with the analysis of the corpus. Moreover, texts from different Arab countries may yield different types of errors, which could then be added to the DECA developed in this study as a standard error classification system, which could then be applied to other Arabic dyslexia corpora.

- As Rello (2014) noted, “Good for dyslexics, useful for all”. The work presented in this thesis is primarily intended for Arabic people with dyslexia, as this field interests the researcher of the thesis. However, this does not prevent the potential for the work to extend to other target groups, such as people with Asperger’s Syndrome.

References

- Aabed, M., Awaideh, S. M., Elshafei, A. M., and Gutub, A. (2007). Arabic diacritics based steganography. In *IEEE International Conference on Signal Processing and Communications, Dubai, United Arab Emirates*, pages 756–759. IEEE.
- Aaron, P. G. (1989). Orthographic systems and developmental dyslexia: A reformulation of the syndrome. In *Reading and Writing Disorders in Different Orthographic Systems*, pages 379–400. Dordrecht. Springer Netherlands.
- AbdelRaouf, A., Higgins, C. A., Pridmore, T., and Khalil, M. (2010). Building a multi-modal Arabic corpus (MMAC). *International Journal on Document Analysis and Recognition (IJDAR)*, 13(4):285–302.
- Aboudan, R., Eapen, V., Bayshak, M., Al-Mansouri, M., and Al-Shamsi, M. (2011). Dyslexia in the United Arab Emirates university—a study of prevalence in English and Arabic. *International Journal of English Linguistics*, 1(2):64.
- Abu-Rabia, S. (2000). Effects of exposure to literary Arabic on reading comprehension in a diglossic situation. *Reading and Writing*, 13(1):147–157.

- Abu-Rabia, S. (2001). The role of vowels in reading semitic scripts: Data from Arabic and Hebrew. *Reading and Writing*, 14(1-2):39–59.
- Abu-Rabia, S. and Awwad, J. S. (2004). Morphological structures in visual word recognition: the case of Arabic. *Journal of Research in Reading*, 27(3):321–336.
- Abu-Rabia, S. and Sammour, R. (2013). Spelling errors’ analysis of regular and dyslexic bilingual Arabic-English students. *Open Journal of Modern Linguistics*, 3(01):58.
- Abu-Rabia, S., Share, D., and Mansour, M. S. (2003). Word recognition and basic cognitive processes among reading-disabled and normal readers in Arabic. *Reading and Writing*, 16(5):423–442.
- Abu-Rabia, S. and Taha, H. (2004). Reading and spelling error analysis of native Arabic dyslexic readers. *Reading and Writing*, 17(7):651–690.
- Abu-Rabia, S. and Taha, H. (2006). Phonological errors predominate in Arabic spelling across grades 1–9. *Journal of Psycholinguistic Research*, 35(2):167–188.
- Abunayyan, I. (2003). Error analysis in spelling. Retrieved from: https://talzahraneikau.edu.sa/Show_Files.aspx?Site_ID=0010492&Lng=AR [Accessed: 20 December 2014].
- Al-Barhamtoshy, H. M. and Motaweh, D. M. (2017). Diagnosis of dyslexia using computation analysis. In *International Conference on Informatics, Health & Technology (ICIHT), Riyadh, Saudi Arabia*, pages 1–7. IEEE.
- Al-kazaz, N. (2018). *Compression-based Methods for the Automatic Crypt-analysis of Classical Ciphers*. PhD thesis, Prifysgol Bangor University.

- Al-kazaz, N. R., Irvine, S. A., and Teahan, W. J. (2018). An automatic cryptanalysis of simple substitution ciphers using compression. *Information Security Journal: A Global Perspective*, 27(1):57–75.
- Al Rowais, F., Wald, M., and Wills, G. (2013). An Arabic framework for dyslexia training tools. In *1st International Conference on Technology for Helping People with Special Needs (ICTHP)*, Saudi Arabia, pages 63–68.
- Al-Sulaiti, L. and Atwell, E. S. (2006). The design of a corpus of contemporary Arabic. *International Journal of Corpus Linguistics*, 11(2):135–171.
- Alamri, M. M. (2013). Investigating dyslexic Arabic text. Master’s thesis, The School of Computer Science, Bangor University.
- Alamri, M. M. and Teahan, W. J. (2017). A new error annotation for dyslexic texts in Arabic. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 72–78. Association for Computational Linguistics.
- Alamri, M. M. and Teahan, W. J. (2019). Automatic correction of Arabic dyslexic text. *Computers*, 8(1):19.
- Alfaifi, A. Y. G. (2015). *Building the Arabic Learner Corpus and a System for Arabic Error Annotation*. PhD thesis, University of Leeds.
- Alhawiti, K. M. (2014). *Adaptive models of Arabic text*. PhD thesis, Prifysgol Bangor University.
- Ali, M. (2011). *Learning difficulties between skills and disorders*. 1 edition. Dar Safa for Publishing & Distributing.
- Alkahtani, S. (2015). *Building and verifying parallel corpora between Ara-*

- bic and English*. PhD thesis, The School of Computer Science, Bangor University.
- Alkanhal, M. I., Al-Badrashiny, M. A., Alghamdi, M. M., and Al-Qabbany, A. O. (2012). Automatic stochastic Arabic spelling correction with emphasis on space insertions and deletions. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(7):2111–2122.
- Almahdawi, A. and Teahan, W. J. (2018). Automatically recognizing emotions in text using prediction by partial matching (PPM) text compression method. In Al-mamory, S. O., Alwan, J. K., and Hussein, A. D., editors, *New Trends in Information and Communications Technology Applications*, pages 269–283, Cham. Springer International Publishing.
- Almeman, K. and Lee, M. (2013). A comparison of Arabic speech recognition for multi-dialect vs. specific dialects. In *The 7th International Conference on Speech Technology and Human-Computer Dialogue (SpeD 2013)*, Cluj-Napoca, Romania, pages 16–19.
- Alnaim, F. (2015). Approaches for identification of dyslexia. *International Journal of Advances in Science Engineering and Technology*, 3(2):67–69.
- Alrabiah, M., Al-Salman, A., and Atwell, E. (2013). The design and construction of the 50 million words KSUCCA. In *Proceedings of WACL’2 Second Workshop on Arabic Corpus Linguistics*, pages 5–8. Lancaster University, UK.
- Alsaleem, S. (2011). Automated Arabic text categorization using SVM and NB. *International Arab Journal of e-Technology*, 2(2):124–128.
- AlShenaifi, N., AlNefie, R., Al-Yahya, M., and Al-Khalifa, H. (2015). A hybrid cascade model for Arabic spelling error detection and correction.

- In *The Second Workshop on Arabic Natural Language Processing, Beijing, China*, pages 127–132. Association for Computational Linguistics.
- Altamimi, M. and Teahan, W. J. (2017). Gender and authorship categorisation of Arabic text from Twitter using PPM. *International Journal of Computer Science & Information Technology (IJCSIT)*, 9(2).
- Altantawy, M., Habash, N., Rambow, O., and Saleh, I. (2010). Morphological analysis and generation of Arabic nouns: A morphemic functional approach. In *The Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).
- Aston, G. and Burnard, L. (1998). *The BNC handbook: exploring the British National Corpus with SARA*. Capstone.
- Attia, M., Pecina, P., Samih, Y., Shaalan, K., and Genabith, J. (2012). Improved spelling error detection and correction for Arabic. *Proceedings of COLING 2012 Posters*, pages 103–112.
- Awad, M. and Khanna, R. (2015). *Efficient learning machines: theories, concepts, and applications for engineers and system designers*. Apress.
- Baker, P. (2006). *Using corpora in discourse analysis*. A&C Black.
- Baraka, R. S., Salem, S., Abu Hussien, M., Nayef, N., and Abu Shaban, W. (2014). Arabic text author identification using Support Vector Machines. *Journal of Advanced Computer Science and Technology Research*, 4(1).
- Baron, A., Rayson, P., and Archer, D. (2009). Word frequency and key word statistics in corpus linguistics. *Anglistik: International Journal of English Studies*, 20(1):41–67.

- Bell, T., Witten, I. H., and Cleary, J. G. (1989). Modeling for text compression. *ACM Computing Surveys (CSUR)*, 21(4):557–591.
- Bell, T. C., Cleary, J. G., and Witten, I. H. (1990). *Text compression*. New Jersey: Prentice-Hall, Inc.
- Bennett, G. R. (2010). *Using corpora in the language learning classroom: corpus linguistics for teachers*. University of Michigan Press, Michigan.
- Berninger, V. W., Nielsen, K. H., Abbott, R. D., Wijsman, E., and Raskind, W. (2008). Writing problems in developmental dyslexia: Under-recognized and under-treated. *Journal of School Psychology*, 46(1):1–21.
- Berninger, V. W. and Wolf, B. J. (2016). *Dyslexia, dysgraphia, OWL LD, and dyscalculia*. Brookes Publishing.
- Biadisy, F., Hirschberg, J., and Habash, N. (2009). Spoken Arabic dialect identification using phonotactic modeling. In *Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages*, pages 53–61, Athens, Greece. Association for Computational Linguistics.
- Biber, D. (1993). Representativeness in corpus design. *Literary and linguistic computing*, 8(4):243–257.
- Bouckaert, R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A., and Scuse, D. (2013). Weka manual for version 3-7-8. Retrieved from: http://statweb.stanford.edu/lpekelis/13_datafest_cart/WekaManual-3-7-8.pdf [Accessed: 02 September 2018].
- Bourassa, D. C. and Treiman, R. (2008). Morphological constancy in spelling: a comparison of children with dyslexia and typically developing children. *Dyslexia*, 14(3):155–169.

- Brill, E. and Moore, R. C. (2000). An improved error model for noisy channel spelling correction. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 286–293, Hong Kong. Association for Computational Linguistics.
- British Dyslexia Association (2007). Definitions. Retrieved from: <http://www.bdadyslexia.org.uk/dyslexic/definitions> [Accessed: 10 May 2014].
- Brosh, H. (2015). Arabic spelling: Errors, perceptions, and strategies. *Foreign Language Annals*, 48(4):584–603.
- Chowdhury, G. G. (2003). Natural language processing. *Annual review of information science and technology*, 37(1):51–89.
- Church, K. W. and Gale, W. A. (1991). Probability scoring for spelling correction. *Statistics and Computing*, 1(2):93–103.
- Cleary, J. and Witten, I. (1984). Data compression using adaptive coding and partial string matching. *IEEE Transactions on Communications*, 32(4):396–402.
- Cleary, J. G. and Teahan, W. J. (1997). Unbounded length contexts for ppm. *The Computer Journal*, 40(2_and_3):67–75.
- Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.
- Comrie, B. (2009). *The world’s major languages*. Routledge.
- Cortes, C. and Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20(3):273–297.

- Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3):171–176.
- Davis, R. D. and Braun, E. M. (1997). *The gift of dyslexia: why some of the brightest people can't read and how they can learn*. Souvenir Press, London.
- Douglas, S. (2012). The intelligent spell checker. Blog. Retrieved from: <https://thecodpast.org/2015/12/the-intelligent-spell-checker/> [Accessed 3 Sep 2015].
- Drucker, H., Wu, D., and Vapnik, V. N. (1999). Support Vector Machine for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054.
- Dumais, S. and Chen, H. (2000). Hierarchical classification of web content. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 256–263, Athens, Greece.
- El-Kourdi, M., Bensaid, A., and Rachidi, T. e. (2004). Automatic Arabic document categorization based on the naïve Bayes algorithm. In *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*, pages 51–58, Geneva, Switzerland. Association for Computational Linguistics.
- Elbeheri, G. and Everatt, J. (2007). Literacy ability and phonological processing skills amongst dyslexic and non-dyslexic speakers of arabic. *Reading and writing*, 20(3):273–294.
- Elbeheri, G., Everatt, J., Reid, G., and al Mannai, H. (2006). Dyslexia

- assessment in Arabic. *Journal of Research in Special Educational Needs*, 6(3):143–152.
- Elshafei, M., Al-Muhtaseb, H., and Al-Ghamdi, M. (2006). Machine generation of Arabic diacritical marks. *MLMTA*, 2006:128–133.
- Fischer, F. W., Shankweiler, D., and Liberman, I. Y. (1985). Spelling proficiency and sensitivity to word structure. *Journal of Memory and Language*, 24(4):423–441.
- Francis, W. N. and Kucera, H. (1979). Brown Corpus Manual. Retrieved from: <http://clu.uni.no/icame/manuals/BROWN/INDEX.HTM> [Accessed: 24 Dec 2018].
- Frank, E. and Bouckaert, R. R. (2006). Naive Bayes for text classification with unbalanced classes. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 503–510. Springer.
- Frank, E., Chui, C., and Witten, I. H. (2000). Text categorization using compression models. Technical report, University of Waikato, Department of Computer Science, Hamilton, New Zealand.
- Frid, A. and Breznitz, Z. (2012). An SVM based algorithm for analysis and discrimination of dyslexic readers from regular readers using ERPs. In *IEEE 27th Convention of Electrical and Electronics Engineers*, pages 1–4, Eilat, Israel.
- Gavioli, L. and Aston, G. (2001). Enriching reality: language corpora in language pedagogy. *English Language Teaching*, 55(3):238–246.
- Ghazaleh, E. B. (2011). A study of developmental dyslexia in middle school foreign language learners in Iran. *Argumentum*, 7:159–169.

- Goulandris, N. E. (2003). *Dyslexia in different languages: Cross-linguistic comparisons*. Whurr Notes.
- Graham, S., Harris, K. R., and Larsen, L. (2001). Prevention and intervention of writing difficulties for students with learning disabilities. *Learning Disabilities Research & Practice*, 16(2):74–84.
- Granger, S. (2003). Error-tagged Learner Corpora and CALL: A Promising Synergy. *CALICO Journal*, 20:465–480.
- Greenbaum, S. and Nelson, G. (1996). The International Corpus of English (ICE) Project. *World Englishes*, 15(1):3–15.
- Grigorenko, E. L. (2001). Developmental dyslexia: An update on genes, brains, and environments. *Journal of Child Psychology and Psychiatry*, 42(1):91–125.
- Guardiola, J. G. (2001). The evolution of research on dyslexia. *Anuario De Psicología*, 32(1):3–30.
- Gupta, N. and Mathur, P. (2012). Spell checking techniques in NLP: a survey. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(12).
- Gutub, A., Elarian, Y., Awaideh, S., and Alvi, A. (2008). Arabic Text Steganography Using Multiple Diacritics. In *International Workshop on Signal Processing and its Applications*.
- Gutub, A., Ghouti, L., Elarian, Y., Awaideh, S., and Alvi, A. (2010). Utilizing diacritic marks for Arabic text steganography. *Kuwait Journal of Science & Engineering (KJSE)*, 37(1):89–109.

- Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic Transliteration. In *Arabic Computational Morphology: Knowledge-based and Empirical Methods*, pages 15–22. Springer Netherlands, Dordrecht.
- Habash, N. Y. (2010). Introduction to Arabic Natural Language Processing. *Synthesis Lectures on Human Language Technologies*, 3(1):9–11.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.
- Hall, W. (2009). *Dyslexia in the primary classroom*. Learning Matters.
- Hamadneh, B. M., Al-Salahat, M. M., Al-Shradgeh, M. T., and Alali, W. A. (2014). Degree of common misspellings of students with learning disabilities. *The International Interdisciplinary Journal of Education (IIJOE)*, 3(6).
- Harrat, S., Abbas, M., Meftouh, K., and Smaïli, K. (2013). Diacritics restoration for Arabic dialect texts. In *INTERSPEECH 2013-14th Annual Conference of the International Speech Communication Association*, pages 1429–1433.
- Hassan, Y., Aly, M., and Atiya, A. (2014). Arabic spelling correction using supervised learning. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 121–126, Doha, Qatar. Association for Computational Linguistics.
- Hiscox, L., Leonavičiūtė, E., and Humby, T. (2014). The effects of automatic spelling correction software on understanding and comprehension in compensated dyslexia: improved recall following dictation. *Dyslexia*, 20(3):208–224.

- Holes, C. (2004). *Modern Arabic: Structures, functions, and varieties*. Georgetown University Press.
- Howard, P. G. (1993). *The design and analysis of efficient lossless data compression systems*. PhD thesis, Brown University.
- Hunston, S. (2002). *Corpora in applied linguistics*. Ernst Klett Sprachen.
- Hunston, S. (2006). Corpus linguistics. In *The Encyclopaedia of Language and Linguistics*, volume 7, pages 215–244. 2 edition. Elsevier.
- International Dyslexia Association (2002). Definition of dyslexia. Retrieved from: <https://dyslexiaida.org/definition-of-dyslexia/> [Accessed: 3 Sep 2015].
- International Dyslexia Association (2012). Dyslexia basics. Retrieved from: <http://dyslexiahelp.umich.edu/sites/default/files/DyslexiaBasic-sREVMay2012.pdf> [Accessed: 24 Dec 2018].
- Islam, A. and Inkpen, D. (2009). Real-word spelling correction using Google Web IT 3-grams. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, volume 3, pages 1241–1249. Association for Computational Linguistics.
- Järvinen, T. (1994). Annotating 200 million words: the Bank of English project. In *Proceedings of the 15th conference on Computational linguistics*, volume 1, pages 565–568. Association for Computational Linguistics.
- Jelinek, F. (1990). Self-organized language modeling for speech recognition. In *Readings in speech recognition*, pages 450–506. Morgan Kaufmann.
- Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In Nédellec, C. and Rouveirol,

- C., editors, *European Conference on Machine Learning: ECML-98*, pages 137–142, Berlin, Heidelberg. Springer. Springer Berlin Heidelberg.
- Johansson, S. (1980). The LOB corpus of British English texts: presentation and comments. *ALLC Journal*, 1:25–36.
- Jurafsky, D. and Martin, J. H. (2018). Speech and language processing. Retrieved from: <https://web.stanford.edu/jurafsky/slp3/ed3book.pdf> [Accessed: 05 October 2018].
- Karim, I., Qayoom, A., Wahab, A., and Kamaruddin, N. (2013). Early identification of dyslexic preschoolers based on neurophysiological signals. In *International Conference on Advanced Computer Science Applications and Technologies*, pages 362–366, Kuching, Malaysia. IEEE.
- Kemp, N., Parrila, R. K., and Kirby, J. R. (2009). Phonological and orthographic spelling in high-functioning adult dyslexics. *Dyslexia*, 15(2):105–128.
- Kernighan, M. D., Church, K. W., and Gale, W. A. (1990). A Spelling Correction Program based on a Noisy Channel Model. In *Proceedings of the 13th conference on Computational linguistics*, volume 2, pages 205–210. Association for Computational Linguistics.
- Khan, R. U., Cheng, J. L. A., and Bee, O. Y. (2018). Machine Learning and Dyslexia: Diagnostic and Classification System (DCS) for Kids with Learning Disabilities. *International Journal of Engineering & Technology*, 7:97–100.
- Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings: the tenth Machine Translation Summit*, volume 5, pages 79–86, Phuket, Thailand.

- Kohli, M. and Prasad, T. (2010). Identifying dyslexic students by using Artificial Neural Networks. In *Proceedings of the World Congress on Engineering*, volume 1, London, UK.
- Kohn, K. (2012). Pedagogic corpora for content and language integrated learning. insights from the backbone project. *The Eurocall Review*, 20(2):3–22.
- Korhonen, T. (2008). *Adaptive spell checker for dyslexic writers*. Springer.
- Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., and Brown, D. (2019). Text classification algorithms: A survey. *Information*, 10(4):150.
- Kukich, K. (1992). Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24(4):377–439.
- Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.
- Larkey, L. S., Ballesteros, L., and Connell, M. E. (2002). Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–282, Tampere, Finland.
- Leahy, M. (2002). Spelling, spelling-checkers and dyslexia. In *Computer Education Society of Ireland CESI Conference*.
- Liensberger, C. (2015). Context sensitive auto-correction. Retrieved from: <https://patents.google.com/patent/US9218333B2/en> [Accessed: 20 Nov 2017].

- Lindgrén, S.-A. and Laine, M. (2011). Multilingual dyslexia in university students: reading and writing patterns in three languages. *Clinical Linguistics & Phonetics*, 25(9):753–766.
- Liu, M., An, Y., Hu, X., Langer, D., Newschaffer, C., and Shea, L. (2013). An Evaluation of Identification of Suspected Autism Spectrum Disorder (ASD) Cases in Early Intervention (EI) Records. In *IEEE International Conference on Bioinformatics and Biomedicine*, pages 566–571, Shanghai, China.
- MacArthur, C. A., Graham, S., Haynes, J. B., and DeLaPaz, S. (1996). Spelling checkers and students with learning disabilities: performance comparisons and impact on spelling. *The Journal of Special Education*, 30(1):35–57.
- MacWhinney, B. (1996). The CHILDES system. *American Journal of Speech-Language Pathology*, 5(1):5–14.
- Majumder, P., Mitra, M., and Chaudhuri, B. (2002). N-gram: a language independent approach to IR and NLP. In *International Conference on Universal Knowledge and Language*, Goa, India.
- Manis, F. R., Szeszulski, P. A., Holt, L. K., and Graves, K. (1990). Variation in component word recognition and spelling skills among dyslexic children and normal readers. In *Reading and its development: Component skills approaches*, pages 207–259. San Diego, CA, US. Academic Press.
- Mars, M. (2016). Toward a Robust Spell Checker for Arabic Text. In *Computational Science and Its Applications – ICCSA*, pages 312–322, Cham. Springer International Publishing.

- McCarthy, M. (2004). *Touchstone: from corpus to course book*. Cambridge University Press.
- McEnery, T. and Wilson, A. (2001). *Corpus linguistics: An introduction*. Edinburgh University Press.
- McEnery, T. and Xiao, R. (2010). What corpora can offer in language teaching and learning. In *Handbook of Research in Second Language Teaching and Learning*, volume 2, pages 364–380. Routledge.
- Mertsalov, K. and McCreary, M. (2009). Document classification with Support Vector Machines. *ACM Computing Surveys (CSUR)*, pages 1–47.
- Meyler, A. and Breznitz, Z. (2003). Processing of phonological, orthographic and cross-modal word representations among adult dyslexic and normal readers. *Reading and Writing*, 16(8):785–803.
- Ministry of Education of Saudi Arabia (2015). Special education teachers’ policy guidelines. Retrieved from: <https://departments.moe.gov.sa/EducationAgency/RelatedDepartments/boysSpecialEducation/Documents/Teacher%20learning%20disabilities%20guide.pdf> [Accessed: 10 Aug 2015].
- Mishra, R. and Kaur, N. (2013). A survey of spelling error detection and correction techniques. *International Journal of Computer Trends and Technology*, 4(3).
- Moats, L. C. (1993). Spelling error interpretation: beyond the phonetic/dysphonetic dichotomy. *Annals of Dyslexia*, 43(1):174–185.
- Moffat, A. (1990). Implementing the PPM data compression scheme. *IEEE Transactions on Communication*, 38(11):1917–1921.

- Mohamad, S., Mansor, W., and Lee, K. Y. (2013). Review of neurological techniques of diagnosing dyslexia in children. In *IEEE 3rd International Conference on System Engineering and Technology (ICSET)*, pages 389–393, Shah Alam, Malaysia.
- Mohammad, A. H., Alwada'n, T., and Al-Momani, O. (2016). Arabic text categorization using Support Vector Machine, naïve Bayes and Neural Network. *GSTF Journal on Computing (JoC)*, 5(1).
- Montgomery, D. J., Karlan, G. R., and Coutinho, M. (2001). The effectiveness of word processor spell checker programs to produce target words for misspellings generated by students with learning disabilities. *Journal of Special Education Technology*, 16(2):27–42.
- Morgan, W. P. (1896). A case of congenital word blindness. *British Medical Journal*, 2(1871):1378.
- Mortimore, T. (2008). *Dyslexia and learning style: a practitioner's handbook*. John Wiley & Sons.
- Mubarak, H. and Darwish, K. (2014). Automatic correction of Arabic text: a cascaded approach. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 132–136.
- Mubarak, H., Darwish, K., and Abdelali, A. (2015). Qcri qalb-2015 shared task: correction of Arabic text for native and non-native speakers' errors. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 150–154.
- Nawar, M. and Ragheb, M. (2014). Fast and robust Arabic error correction system. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 143–147.

- Nawar, M. and Ragheb, M. (2015). Cufe qalb-2015 shared task: Arabic error correction system. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 133–137.
- Nelson, M. and Gailly, J.-L. (1996). *The data compression book*. 2 edition. M & T Books New York.
- Nicholls, D. (2003). The Cambridge learner corpus: Error coding and analysis for lexicography and ELT. In *Proceedings of the Corpus Linguistics*, volume 16, pages 572–581.
- Noaman, H. M., Sarhan, S. S., and Rashwan, M. (2016). Automatic Arabic spelling errors detection and correction based on confusion matrix-noisy channel hybrid system. *Egyptian Computer Science Journal*, 40(2).
- Norvig, P. (2009). Natural language corpus data. In *Beautiful data: the stories behind elegant data solutions*, pages 219–242. O’Reilly Media.
- Nunez, P. L. and Srinivasan, R. (2006). *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, USA.
- Pedler, J. (2007). *Computer correction of real-word spelling errors in dyslexic text*. PhD thesis, Birkbeck College, University of London.
- Perera, H., Shiratuddin, M. F., and Wong, K. W. (2016). A review of electroencephalogram-based analysis and classification frameworks for dyslexia. In *Neural Information Processing*, pages 626–635, Cham. Springer International Publishing.
- Prevett, P., Bell, S., and Ralph, S. (2013). Dyslexia and education in the 21st century. *Journal of Research in Special Educational Needs*, 13(1):1–6.

- Pustejovsky, J. and Stubbs, A. (2012). *Natural language annotation for machine learning*. O'Reilly Media, Inc.
- Rauschenberger, M., Rello, L., Füchsel, S., and Thomaschewski, J. (2016). A language resource of German errors written by children with dyslexia. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia. European Language Resources Association (ELRA).
- Reid, G. (2010). *Dyslexia*. 2 edition. Bloomsbury Publishing.
- Rello, L. (2014). *A text accessibility model for people with dyslexia*. PhD thesis, Department of Information and Communication Technologies, University Pompeu Fabra.
- Rello, L., Baeza-Yates, R., Saggion, H., and Pedler, J. (2012). A first approach to the creation of a Spanish corpus of dyslexic texts. In *Natural Language Processing for Improving Textual Accessibility (NLP4ITA)*, Istanbul, Turkey.
- Rello, L. and Ballesteros, M. (2015). Detecting readers with dyslexia using machine learning with eye tracking measures. In *Proceedings of the 12th Web for All Conference*, Florence, Italy. ACM.
- Rello, L., Ballesteros, M., and Bigham, J. P. (2015). A spellchecker for dyslexia. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, pages 39–47.
- Samanta, P. and Chaudhuri, B. B. (2013). A simple real-word error detection and correction using local word bigram and trigram. In *Proceedings of the 25th Conference on Computational Linguistics and Speech Processing (ROCLING)*, pages 211–220, Kaohsiung, Taiwan.

- Sayood, K. (2017). *Introduction to data compression*. 5 edition. Morgan Kaufmann.
- Shaalán, K. F., Attia, M., Pecina, P., Samih, Y., and van Genabith, J. (2012). Arabic word generation and modelling for spell checking. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 719–725, Istanbul, Turkey. European Languages Resources Association (ELRA).
- Shalom, C. (1997). That great supermarket of desire: Attributes of the desired other in personal advertisements. In *Language and desire: Encoding sex, romance and intimacy*, pages 186–203. Routledge.
- Simons, G. F. and Fennig, C. D. (2018). Ethnologue: Languages of the world. Retrieved from: <https://www.ethnologue.com/statistics/size> [Accessed: 13 Dec 2018].
- Snowling, M. J. (2012). Changing concepts of dyslexia: nature, treatment and comorbidity. *Journal of Child Psychology and Psychiatry*, 53(9):e1–e3.
- Snowling, M. J. (2013). Early identification and interventions for dyslexia: a contemporary view. *Journal of Research in Special Educational Needs*, 13(1):7–14.
- Snowling, M. J., Muter, V., and Carroll, J. (2007). Children at family risk of dyslexia: a follow-up in early adolescence. *Journal of Child Psychology and Psychiatry*, 48(6):609–618.
- Sornil, O. and Chaiwanarom, P. (2004). Combining Prediction by Partial Matching and Logistic Regression for Thai Word Segmentation. In *COLING 2004: Proceedings of the 20th International Conference on Compu-*

- tational Linguistics*, pages 1208–1213, Geneva, Switzerland. Association for Computational Linguistics.
- Tamboer, P., Vorst, H., Ghebreab, S., and Scholte, H. (2016). Machine Learning and Dyslexia: Classification of Individual Structural Neuroimaging Scans of Students with and without Dyslexia. *NeuroImage: Clinical*, 11:508–514.
- Teahan, W. J. (1998). *Modelling English text*. PhD thesis, University of Waikato.
- Teahan, W. J. (2000). Text classification and segmentation using minimum cross-entropy. *Content-Based Multimedia Information Access*, 2:943–961.
- Teahan, W. J. (2018). A Compression-based Toolkit for Modelling and Processing Natural Language Text. *Information*, 9(12):294.
- Teahan, W. J. and Cleary, J. G. (1997). Models of English Text. In *Proceedings of Data Compression Conference*, pages 12–21, Snowbird, UT, USA. IEEE.
- Teahan, W. J. and Harper, D. J. (2003). Using Compression-based Language Models for Text Categorization. In Croft, W. B. and Lafferty, J., editors, *Language Modeling for Information Retrieval*, pages 141–165. Springer Netherlands.
- Teahan, W. J., Inglis, S., Cleary, J. G., and Holmes, G. (1998). Correcting English text using PPM models. In *Data Compression Conference*, pages 289–298. IEEE.
- Teahan, W. J., Wen, Y., McNab, R., and Witten, I. H. (2000). A compression-based algorithm for Chinese word segmentation. *Computational Linguistics*, 26(3):375–393.

- Thomas, C., Keselj, V., Cercone, N., Rockwood, K., and Asp, E. (2005). Automatic detection and rating of dementia of Alzheimer type through lexical analysis of spontaneous speech. In *IEEE International Conference Mechatronics and Automation*, volume 3, pages 1569–1574, Niagara Falls, Ont., Canada.
- Vasa, K. (2016). Text classification through statistical and machine learning methods: a survey. *International Journal of Engineering Development and Research*, 4:655–658.
- Vellutino, F. R., Fletcher, J. M., Snowling, M. J., and Scanlon, D. M. (2004). Specific reading disability (dyslexia): what have we learned in the past four decades? *Journal of Child Psychology and Psychiatry*, 45(1):2–40.
- Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.
- W3Techs (2013). Historical trends in the usage of character encodings for websites. Retrieved from: https://w3techs.com/technologies/history__overview/character__encoding [Accessed: 30 September 2017].
- Xiao, R. (2010). Corpus creation. In *Handbook of Natural Language Processing*. Boca Raton, FL, 2 edition. CRC Press.
- Zaghouani, W., Zerrouki, T., and Balla, A. (2015). SAHSON QALB-2015 shared task: a rule-based correction method of common Arabic native and non-native speakers’ errors. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 155–160.
- Zainuddin, A., Lee, K. Y., Mansor, W., and Mahmoodin, Z. (2016). Optimized KNN classify rule for EEG based differentiation between capable

- dyslexic and normal children. In *IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, pages 685–688, Kuala Lumpur, Malaysia.
- Zainuddin, A., Mansor, W., Lee, Y. K., and Mahmoodin, Z. (2018). Performance of Support Vector Machine in Classifying EEG Signal of Dyslexic Children using RBF Kernel. *Indonesian Journal of Electrical Engineering and Computer Science*, 9(2):403–409.
- Zamora, E., Pollock, J. J., and Zamora, A. (1981). The use of trigram analysis for spelling error detection. *Information Processing & Management*, 17(6):305–316.
- Zhang, W., Yoshida, T., and Tang, X. (2008). Text classification based on multi-word with support vector machine. *Knowledge-Based Systems*, 21(8):879–886.
- Zhou, L., Baughman, A., Lei, V., Lai, K., Navathe, A., Chang, F., Sordo, M., Topaz, M., Zhong, F., Murralli, M., Navathe, S., and Rocha, R. (2015). Identifying patients with depression using free-text clinical documents. *Studies in Health Technology and Informatics*.

Appendix 1: Ministry of Education Authorisation Letter

المملكة العربية السعودية
وزارة التربية والتعليم
إدارة التخطيط والتطوير

الرقم :
التاريخ :
المستلمة من :

وزارة التربية والتعليم
Ministry of Education

"إفادة"

الموضوع : الموافقة على تطبيق أبحاث دراسية على مدارس وإدارات تابعة لإدارة التربية والتعليم بمنطقة الرياض

اسم الباحث	مها مرزوق العمري .
الكلية / الجامعة	جامعة بانهور / بريطانيا .
الغرض من الدراسة	متطلب للحصول على درجة / الدكتوراه .
مجال الدراسة والعينة	عينة من طالبات سعوديات تتعلم في المملكة العربية السعودية .

سعادة المحقق الثقافي السعودي في / بريطانيا

حفظه الله

السلام عليكم ورحمة الله وبركاته

ويعد ..

بناءً على تعميم معالي وزير التربية والتعليم رقم ٥٥/٦١٠ وتاريخ ١٤١٦/٩/١٧هـ بشأن تفويض الإدارات العامة للتربية والتعليم بإصدار خطابات السماح للباحثين بإجراء البحوث والدراسات . وبناءً على تفويض مدير عام إدارة التربية والتعليم إدارة التخطيط والتطوير ذي الرقم ١١/٣٣٦٧٤٨٢٣ والتاريخ ١٤٣٣/٤/١٤هـ بشأن تسهيل مهام الباحثين والباحثات . نفيدكم أنه لا مانع من تطبيق الدراسة على نطاق مجال العينة المحددة / التابعة لإدارة التربية والتعليم بمنطقة الرياض . مع ملاحظة أن الباحث / ة يتحمل كامل المسؤولية المتعلقة بمختلف جوانب البحث . ولا يعني سماح الإدارة العامة للتربية والتعليم موافقتها بالضرورة على مشكلة البحث أو على المنهج والأساليب المستخدمة في دراستها ومعالجتها . وبناءً على طلبها تم منحها الإفادة .

شاكرين طيب تعاونكم .

مدير إدارة التخطيط والتطوير
١٠/١٤
عنه بدرية الحميد
سعود بن راشد آل عبد اللطيف

وزارة التربية والتعليم
إدارة التخطيط والتطوير
إدارة الشؤون العامة

في قسم الدراسات والبحوث .

نموذج ١/٦

البريد الإلكتروني : planingm@moe.gov.sa / al-takhteet@hotmail.com هاتف ٤٠٢٠٤٣٨ - فاكس ٤٠٢٧٠٨٨ - سترايل ٤٠٥٩٥٠٠ - الرياض ١١٥١٤

Appendix 2: Participant Consent Form

Participant Consent Form

To be filled in by teacher/ parents acting on behalf of students

Researcher's name Maha Marzouq Alamri

The researcher named above has briefed me to all of the information of her research. The research is focused on students with learning difficulties. In order to complete the research, the researcher needs to obtain a copy of student/child's writing either through the student's book or the spelling book.

All information is confidential, and the student's name or school's name will not be disclosed. Only the gender and age of the student will be mentioned, and the text will be used for scientific publication. If you agree to voluntarily take part in this project, I would like you to sign this consent form. If you have any questions about the project or any further information, please contact the researcher via email (maha.alamri@bangor.ac.uk).

Signature:

Appendix 3: Form That Was Completed by Participant with Dyslexia

Age:

Year:

What is your favourite hobby?

What do you want to be when you grow up?

What would you take with you if you were to go to an island?

What is your wish you want to have in the future?

Write about your school, friends or travel

Appendix 4: Arabic Transcription Standard

There is no standard practice for transcribing Arabic from a handwritten format into a computerised form. Therefore Alfaifi (2015) developed a series of standards for achieving a high level of consistency during transcription as follows:

1. Any struck-out texts should be excluded.
2. If there is a correction above a non-struck out word, the corrected form is transcribed.
3. When there is a doubtful form of a character, the form closest to the correct form is transcribed.
4. If there is an overlap between handwritten characters, which cannot be transcribed, the closest possible form is selected.
5. If a writer forgot to add a character's dot(s) whether above or below, it should be transcribed as written by the learner, unless it is not possible (e.g. if there is no equivalent character on the computer).
6. A new line (paragraph) should be inserted only when the learner has clearly done so. Examples include if there is a clear space at the end of a line (whether there is a period or not) or if there is a clear space at the beginning of a new line with a period at the end of the previous paragraph. Other instances, such as ending a line with a period but with no clear space at the end or at the beginning of the new line, are considered as a single paragraph.
7. Any identifying information (e.g. learner's name, contacts, postal address, emails, etc.), which were replaced in the PDF sheet with "per-

sonal information deleted”, should be transcribed as in the computerised text. Other non-personal information can be left such as class, name of school, city, country, religion, culture, etc.

8. Any shape, illustration, or ornamentation drawn by the learner on the sheet is excluded.
9. Texts with no titles are given (text with no title) in the title field.
10. Any text format is excluded such as underlined words or sentences.
11. Unknown words or phrases are replaced with (unknown word), or (unknown phrase).

Appendix 5: BDAC Examples

غرفتن
جزرن
مقاطعة
وصلة
كوثرو
نممل
ركظ
القلام
صلات العاش الربع ركعة
تعايش الصلحقات في البر و البحر
أصبح أبن ماجدي ملاحن ماهرن
أنا أحب القراءة و الكتابة
عاد ساعيد من الصفر مسرورا
خالد يحب بلادة وأصله و أصحابه
في المدرسة أتعلم القراءة و الكتابة
الله الذي خلق الشمس والقمر
دع عمرن أخه خلدن ليعب معه بقطره الجد بحقت المزل لكنه سقط من يده فتقسر وثقة أجز لصغرة فقدم اعتذاره لعمرن
البط من الطيور المائية وجسمه مهيئ لذلك فتقدم هو قربة من نهاية جسمه مم يساعده على السباحة في الماء بغير تباطئه
والبط ممثلي الجسمي يقضي معظم وقتهم في السباحة
في عهد الخليفة أبي بكر الصديق أصاب الناس جفاف والجوع شديدا فلم يذوق بهم الأمر ذهبوا إلى مجلس الخليفة أبي
بكر الصديق وقال يا خليفة رسول الله قد أدرك الناس من الهلاك فالسما لم تمطر و الأرض لم تثبت و ساد الجوع وعم
الفقر فماذا نفعل
العلم كسراج من مره به اسقبتس من العلم وصاحب العلم ذو الخبرة ودرايها بها يعلم به و العلم نور و الجهلو ظلم يضل
على صاحبه فالعلم يكون بالدراسة و الممارسة و التعلم مع الاشخاص و البيات و الات المتنوعة ينفع صاحبه من وحوله
فهذا يبذر لهذا قمحا يأكله وهذا يعمل لهذا ثوبا يلبسه وهذا يصنعه لهذا بيتا يساكنه وهذا ينجز لهذا بابا يغلقه على بيته
وغير ذلك مما لا يكاد يدركه العدد من الصناعات و الحاجات لانه ليس في اصطاعات انسان واحد أن يكون فلاحا نساجا
بناء نجارا

Appendix 6: DECA Second Evaluation

May the peace, mercy and blessings of Allah be upon you...

This project includes texts written by students with dyslexia that contains their errors. These errors will then be analysed and classified in order to create tools for supporting them. For the successful completion of this project in a thought-out and integrated manner, I will need the assistance of specialists in the field of learning difficulties.

After reading and analysing their errors, a classification table was then produced containing all dyslexia errors as well as a tag for each error.

Accordingly, I hope you could lend a helping hand by looking at the table and answering the questions below.

Thank you.

Your cooperation is very much appreciated.

Maha Alamri
PhD student – UK
maha.alamri@bangor.ac.uk

الرمز - Tag	أنواع الأخطاء - Error Type	الفئة - Category
<HH>	الهمزة على السطر - Hamza on Line	الهمزات Hamza
<HA>	الهمزة على الألف - Alif Hamza Above	
<HB>	الهمزة تحت الألف - Alif Hamza Below	
<HY>	الهمزة على الياء - Ya Hamza Above	
<HW>	الهمزة على الواو - Waw Hamza Above	
<OT>	لم تذكر في الهمزات - Other	
<AA>	مد الألف - Alif Madd	المدود Almadd
<AW>	مد الواو - Waw Madd	
<AY>	مد الياء - Ya Madd	
<TM>	لم تذكر في المدود - Other	
<CT>	بين التاء المفتوحة والتاء المربوطة - Confusion in Tah and Tah Marbuta/Hah	الخط Confusion
<CH>	بين الهاء والتاء المربوطة - Confusion in Hah and Tah Marbuta	
<CA>	بين الألف المدودة والألف المقصورة - Confusion in Alif and Alif Maksura	
<CD>	بين الظاء والضاد - Confusion in Dha and Tha	
<CV>	الخلط بين حروف متشابهة - Confusion in similar letters	
<OM>	لم تذكر في الخط - Other	
<DN>	نون مكان التنوين - N in Tanwin	الحركات Diacritics
<DW>	واو مكان الضمة - W in Damma	
<DY>	ياء مكان الكسرة - Y in Kasra	
<OD>	لم تذكر في الحركات - Other	
<FW>	وصل ماحقه الفصل من الحروف او فصل ماحقه الوصل من الحروف - Word boundary errors	شكل الكلمة Form
<FM>	أخطاء متعددة - Multi Errors	
<FR>	تكرار الحروف - Repeated Letters	
<OF>	لم تذكر في الشكل - Other	
<MO>	حذف - Omission	الاطعاء الشائعة Common errors
<MA>	إضافة - Addition	
<MS>	تبديل - Substitution	
<MT>	تحويل - Transposition	
<DD>	عدم القدرة على التفريق بين حروف متشابهة لفظا مختلفة شكلا - Different Graphemes, Same Phonetics	الاختلافات Differences
<DF>	عدم القدرة على التفريق بين شكل الحرف إذا كان في بداية الكلمة أو وسطه أو نهايته - Form of the letter in the Beginning, Middle or End	
<DI>	كتابة بناء على اللهجة المحلية - Local language	
<DS>	كتابة كلمة مشابهة للمعنى - Writing a word that is Similar to the Meaning	
<OI>	لم تذكر في الاختلافات - Other	
<WM>	مرآة - Mirror	طريقة الكتابة Writing method
<WL>	كتابة من اليسار لليمين - Left to Right	
<OW>	لم تذكر في الكتابة - Other	
<LS>	لام الشمسية - Sun Letter	حروف تكتب ولا تنطق أو العكس Letter written but not pronounced or vice versa
<LM>	إدخال اللام على ما فيه (ال) - Adding letter (L) to words start with letter (AL)	
<LA>	ألف بعد واو الجماعة - Alif Fariqa	
<LL>	(لاكن - لاكنها ...) - (Lakn ...)	
<LH>	(هاذا - هاذ - هاذان) - (Hada ...)	
<LT>	(التي) - (Ality)	
<LD>	(الذي) - (Alldhy)	
<LK>	(ذلك - بذلك ...) - (Dahlk ...)	
	لم تذكر في حروف - Other	
<OT>	لم تذكر في أي مجموعة - Other	أخرى - Other

- After looking at the classification table and taking a general idea about the errors, is it possible to classify the errors in the table below? This is by placing the appropriate tag for each error based on the symbols at the top.

Examples	Error	Tag	How you can assess the suitability of the error type for errors?			How can you assess the easiness of identifying the tag to the error type?			
			Very suitable	Suitable	Not suitable	Very easy	Easy	Difficult	Did not identify it.
ذهبت هنادون إلى الحديقة	هنادون								
أقر في المصحف	أقر								
خالد تلميذ سبور	سبور								
أرسل محمد رسالت	رسالت								
المدرسة هي التي جمعنا	التي								

- Do you think that the above classification table needs any addition, deletion, or merging of categories?
 - 1- Yes.....
 - 2- No.
- Do you think that each type is placed under the appropriate category?
 - 1- Yes.
 - 2- No.....

- **What is your assessment of how the errors are categorised? For example, errors that include a Hamza are placed under the "Hamza" category and so on.**

1. Convenient.
2. Inappropriate.
3. Needs modification. Please write it below:

.....
.....

- **Is the classification clear and understandable?**

1. Yes.
2. Somewhat.
3. No.

- **Could you please give an overall assessment of this classification?**

.....
.....

Appendix 7: BNDAC Examples

تفائل

رجاء

تشاؤم

إمراة

جائع

قريبا

شيء

ماء

كيسا

إن الله يعلم غيب السموات والأرض والله بصير بما تعملون

العمل عبادة العمل عبادة وهو طريق وقد أمر الله عز وجل بالعمل

المعلم هذا هو التصرف سليم يابدر وليحضر كل منكما أدواته وكاملة

خالد نعم لقد علمني أن قو لصدق دئما فرسلو على الله عليه وسلم قدوتي في الصدق

مدينتان مقدستان في وطني مدينتان مقدستان هما مكة المكرمة و المدينة المنورة

فواز لقد ذهب الغضب يلبي واعدك أن أتقد برساً صلي عليه وسلم ولا أتجج معا احد بعد اليماء

طلب المعلم من التلاميذ التحدث إلى زملائهم عن المصايف الجبلية في المملكة العربية السعودية بدأ عماد الحديث قائلاً من المدن التي زرتها و أعجبتني

وصلت الزائر إلى منزل وفاء وقرعن بابة بهدوء سلمن عليها ودعون لها بالشفاء العاجل وجلسن قليلاً ثم انصرفن

محبة الجار يحكى أن تاجرا ورث دارا عن أبيه فكان يحبها كثيرا ويحافظ عليها ولم يفكر يوما في بيعها أو هجرها ولكن لما كسدت تجارته وتراكمت عليه الديونا عرضها للبيع و حدد مبلغا كبيرا من المال ثمنها لها

الضيف الصغير قال احمر قضيت ليلة أمس انتظر عودة أمي التي ذهبت بصحبة والدي إلى المستشفى تضع مولودها الجديد و في الصباح نبهني من نومي صوت خيل إلي اني كنت اسمعه منذ مدة اثناء نومي فتحت عيني وسرعان ما نقرت وهرولت باتجاه غرفة نوم والدي قرعت الباب و استأذنت ثم دخلت كانت امي تحمل بين ذراعيها الحائيتين ذلك المولود الذي انتظرناه طويلا

ماذهبت يوما إلى القرية إلا رجعت مبتهجا أغبط الفلاحين فهم أساس الحياة و عمادها من عرقهم وتعبد أيديهم نأكل و يأكلون الفلاحون متعاونون متضامنون فهذا يبذر الحبة وذاك يسقي الزرع و آخر يجني الثمر و إذا بنا أخدمهم بيتا أسرع المجاورين لمساعدته و إذا انن المؤذن للصلاة ترك الفلاحون عمالهم و اتجهوا إلى عبادة الله في خشوع لأنهم يعملون لدنياهم و اخرهم وقد شجعت الدولة الفلاحين و بنت لهم المدارس قرب منازلهم و مزارعهم و أمنت لهم الخدمات التي امننتها لغيرهم من المواطنين فتحية إلى الفلاحين و تحية إلى سواعدهم التي تنتج الخير

Appendix 8: Sahah Output

Raw Input Text	Sahah Output text
أهتديتو	أهتديت
مقاطعة	مقاطعة
وصلة	وصلت
فرحة	فرحت
كوثرو	كوثرو
ركض	ركض
فوائد	فوائد
الثقافة	الثقافة
جزرن	جزر
اسعة	ساعة
الذين	الذين
الاساسي	الاساسي
أرسم علا اروق	أرسم على أرواق
هاذا شارع وسع	هذا شارع واسع
حقيقة مزجت بلخيال	حقيقي مزجت بالخيال
العاصفیر ألوانها جمیل	العصافير ألوانها جميلة
أنا أعبد الله وحده لا شريك لهو	أنا أعبد الله وحده لا شريك له
استيقظ محمد بكرا إلى مدرسته	استيقظ محمد بكرا إلى مدرسته
أنا أصلي كل يوم خمس صلوات	أنا أصلي كل يوم خمس صلوات
أصبح أين ماجدي ملاحن ماهرن	أصبح أين ماجدي ملاحا ماهر
الله العظيم هو الذي خلق كل ذلك	الله العظيم هو الذي خلق كل ذلك
تحدث في الحفل من الأطفال خالد ثم زيد	تحدث في الحفل من الأطفال خالد ثم زيد
قال المعلم أن التعاون ثمرة من ثمرات الإيمان	قال المعلم أن التعاون ثمرة من ثمرات الإيمان
ما أجمل الوطن في عيون أصحابه وما أقرب به إلى نفوسهم	ما أجمل الوطن في عيون أصحابه وما أقرب به إلى نفوسهم
دخلت هند الفصل مسرعة حتى تتمكن من حضور الدرس و تستطيع فهمه	دخلت هند الفصل مسرعة حتى تتمكن من حضور الدرس و تستطيع فهمه
عندما تولا عمر ابن عبد العزيز الخلافة أصبحت الوفود تجي إليه تباعا مهينة	عندما تولى عمر ابن عبد العزيز الخلافة أصبحت الوفود تجي إليه تباعا مهينة
الأمومة عاطفت نبيلت أودعها الله في قلب الكائنات الحية جميعا مين إنسان وحيوان وطير	الأمومة عاطفة نبيلة أودعها الله في قلب الكائنات الحية جميعا من إنسان وحيوان وطير
أيها التلميذ اذا اتصفت بالجدية و اخلصت في عملك و اتقنته و عملت بحزم وثبات ولم تضع الوقت حقت اعلى الامنيات	أيها التلميذ إذا اتصفت بالجدية و أخلصت في عملك و أتقنته و عملت بحزم وثبات ولم تضع الوقت حقت أعلی الامنیات
الفلاح رجل نشيط يسكن القرى يحب الخير ويكره الأذى لأي فرد من أفراد قريته يسرع الخطى إلى حقله ومزرعته وبذلك يحقق المنة	الفلاح رجل نشيط يسكن القرى يحب الخير ويكره الأذى لأي فرد من أفراد قريته يسرع الخطا إلى حقله ومزرعته وبذلك يحقق المنة
هذه حديقة جميلة وذلك غزل يجري وتلك دجاجة تمشي فوق السور وهذا خروف يقفز هنا وهناك وهذه بقرة تأكل العشب ما أجمل هذا المكان الله العظيم هو الذي خلق كل ذلك	هذه حديقة جميلة وذلك غزل يجري وتلك دجاجة تمشي فوق السور وهذا خروف يقفز هنا وهناك وهذه بقرة تأكل العشب ما أجمل هذا المكان الله العظيم هو الذي خلق كل ذلك
لا شك أن كل عالم مخترع كان يوما ما فتى صغيرا في مثل سنك وبلد والمثابرة وهدى من الله استطاع أن يحقق المني فاذا أردت أن تكون مثله وتحقق العل لك وللناس فحث الخطى نحو العلم والبحث منذ الصبا	لا شك أن كل عالم مخترع كان يوما ما فتى صغيرا في مثل سنك وبلد والمثابرة وهدى من الله استطاع أن يحقق المني فاذا أردت أن تكون مثله وتحقق العل لك وللناس فحث الخطى نحو العلم والبحث منذ الصبي