

# **Automatic Framework to Aid Therapists to Diagnose Children who Stutter**



**Sadeen Alharbi**

Department of Computer Science

The University of Sheffield

This dissertation is submitted for the degree of

*Doctor of Philosophy*

Speech and Hearing Research Group

October 2018



I would like to dedicate this thesis to my parents and my beloved husband for their love,  
support and encouragement... . . .



## **Declaration**

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Sadeen Alharbi

October 2018



## **Acknowledgements**

Through this PhD journey, I have been able to meet many challenges and achieved my aims through the invaluable support afforded by different people. This thesis would not have been accomplished without the endless support and patience of my first supervisor Dr. Anthony Simons, who always supported and encouraged my ideas. His suggestions and guidance made me a much better person, student and researcher. I would also like to thank my secondary supervisors, Prof. Phil Green and Prof. Roger Moore, for their helpful comments and suggestions regarding my work through each panel meeting. I would like to especially thank Prof. Green for his help and guidance on each publication and supporting me through this journey. Many thanks to Prof. Shelagh Brumfitt from the Human Communication Sciences Department, as I never would have finished the medical side without her invaluable knowledge. I was honoured to work with this exceptional supervisory team, from whom I learned so much and whom I look to as role models. I am thankful to my examiners Dr. Heidi Christensen and Prof. Martin Russell for valuable comments and useful discussion during the final viva of my defence. Special thanks to Madina Hasan, who helped me a lot through my different struggles. My sincere thanks to my colleagues in the Speech and Hearing Lab for their continuous assistance and friendship, especially Mashaël, Rabab, Najwa, Lubna, Eidah, Nazrina, Asif, Gerado and Erfan Lowaimi for their invaluable and incredible support.

I would never have reached this stage without my family. Thank you for keeping me strong throughout this journey. My beloved husband Moayad, your unconditional love, support and understanding underpinned my persistence on this journey, which made the

completion of this thesis possible. My princess Jana and my prince Mohammed, seeing your eyes and hearing your giggles after a long day of work made a big difference. Joanne, my dear baby your blessing soul gave me the strength to finish and success in this journey. My dear parents Fatimah and Sulaiman, thanking you is not enough. I would be neither who nor what I am without you. Thank you to my beloved brothers and sisters, who never hesitate to offer help and warm welcomes, especially my caring sister Samar for her ‘around the clock’ help and advice, even with her busy schedule. Many thanks to Prof. Sami Alwakeel for his incredible support, which has never been forgotten throughout my studies, since I started my bachelor’s degree.

Finally, the work reported in this thesis would not have been possible without the financial support of King Saud University, to which I am grateful.



## Abstract

This thesis studies the feasibility of developing an automated framework that provides an indication of the severity level of stuttering for children who stutter. Diagnosing this condition early is extremely important to ensure adequate therapeutic treatment while a child is still young. However, correct diagnoses depend heavily on the availability of expert therapists and on their painstaking manual work to record, transcribe and then count different kinds of speech disfluency. Where such expertise is rare, an automated framework would be immensely helpful.

The main challenge facing the development of such a system is the scarcity of available training data. Whereas speech corpora of children's speech are limited, corpora of children's stuttering speech are extremely limited, such that certain direct machine-learning techniques could not be used, for lack of training data. Furthermore, the best available stuttering data set is not fully transcribed.

Our proposed solution deploys a number of approaches that make best use of all the available data. We combine a standard children's corpus (PF-Star) with a small corpus of stuttering speech (UCLASS), the latter which we transcribe. We focus on automatic speech recognition (ASR) methods to decode stuttering utterances, making best use of frequency information to segment the speech into words and part-words. We compare two methods, one based on training a conventional ASR whose language model is augmented with extra stuttering events, such as pseudo-words and repetitions; and the other based on task-specific lattices derived from the original reading prompt, with manually-tuned stuttering arcs. We then focus on autocorrelation methods to detect cases of prolongation, making best use of time information, in relation to the subject's speaking rate. We also investigate a number of approaches to identifying different kinds of stuttering event in transcriptions (both manual and produced by ASR), comparing conditional random field (CRF) and bi-directional long short-term memory (BLSTM) detectors. Finally, we use an algorithm based on Guitar, Yairi and Ambrose's classification of stuttering severity, to partition all subjects into three classes: normal disfluency, borderline stuttering and beginning stuttering. The resulting diagnoses were evaluated by comparing them against diagnoses made by two UK-registered speech language therapists, and ranged from 72% to 100% accurate (after tuning).

The benefits of the work include: supporting speech therapists by automatically transcribing recorded sessions, identifying stuttering events and providing an accurate early indication of a diagnosis. This could help children receive suitable therapy, commensurate with the severity of their stuttering. If deployed in the Cloud, this work could help in the remote diagnosis of children in areas where expertise is limited. In this setting, research benefits would include the further collection of stuttering data.



# Contents

<b>List of Figures</b>	<b>xvii</b>
<b>List of Tables</b>	<b>xxi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Research focus . . . . .	3
1.3 Major contributions . . . . .	6
1.4 Thesis overview . . . . .	8
1.5 Published work . . . . .	10
<b>2 What Is Stuttering?</b>	<b>13</b>
2.1 Introduction . . . . .	13
2.2 Nature of stuttering . . . . .	14
2.2.1 Overview . . . . .	14
2.2.2 Impact of stuttering . . . . .	15
2.2.3 Etiology . . . . .	17
2.2.4 Disfluency characteristics of early stuttering . . . . .	19
2.2.5 Risk factors . . . . .	22
2.2.6 Identifying types of stuttering . . . . .	23
2.3 Assessment and diagnosis of stuttering . . . . .	26

2.3.1	The need for background and case history . . . . .	26
2.3.2	Assessment approach . . . . .	27
2.3.3	Stuttering severity . . . . .	28
2.3.4	Disagreement on various assessments of stuttering . . . . .	30
2.4	Treatment of stuttering . . . . .	31
2.4.1	Treatment approaches . . . . .	31
2.4.2	Different opinions on stuttering treatments . . . . .	33
2.5	Summary . . . . .	34
<b>3</b>	<b>Automatic Stuttering Recognition Background</b>	<b>37</b>
3.1	Introduction . . . . .	37
3.2	Direct classification approach . . . . .	39
3.2.1	Artificial Neural Networks (ANN) . . . . .	39
3.2.2	Support Vector Machine (SVM) . . . . .	42
3.2.3	Linear Discriminant Analysis (LDA) and k-Nearest-Neighbour (k-NN)	43
3.2.4	Vector Quantization (VQ) framework, End-Point Detection (EPD) and Dynamic Time Warping (DTW) . . . . .	44
3.3	ASR based approach . . . . .	45
3.3.1	ASR overview . . . . .	46
3.3.2	Applying ASR for classification task . . . . .	56
3.3.3	Applying ASR for detection task . . . . .	57
3.4	Summary . . . . .	58
<b>4</b>	<b>Word Level Transcription and Annotation of Stuttering Data</b>	<b>61</b>
4.1	Introduction . . . . .	62
4.2	UCLASS: Children corpus overview . . . . .	63
4.3	UCLASS: Contributions . . . . .	65

---

4.3.1	Overview . . . . .	65
4.4	Transcription scheme . . . . .	67
4.4.1	Gold standard . . . . .	69
4.5	Annotation scheme . . . . .	69
4.5.1	Gold standard . . . . .	71
4.6	Agreement Measure: Cohen's kappa coefficient . . . . .	72
4.7	Summary . . . . .	74
<b>5</b>	<b>Automatic Stuttering Recognition</b>	<b>75</b>
5.1	Introduction . . . . .	75
5.2	Automatic speech recognition (ASR) approaches for stuttering events detection	77
5.2.1	Previous attempts to use ASR for stuttering . . . . .	78
5.2.2	Stuttering dataset of children's read speech . . . . .	79
5.2.3	Augmentation of stuttering events in LM approach . . . . .	82
5.2.4	Task-oriented lattice approach . . . . .	85
5.3	Summary . . . . .	90
<b>6</b>	<b>Results of the Experiments with the Proposed ASR Approaches</b>	<b>91</b>
6.1	Introduction . . . . .	91
6.2	Summary of building ASR-baseline process . . . . .	92
6.2.1	Acoustic modeling . . . . .	92
6.2.2	Language modelling and pronunciation dictionary . . . . .	93
6.2.3	Pilot study . . . . .	94
6.3	Augmentation of stuttering events in LM approach . . . . .	95
6.3.1	Acoustic modeling . . . . .	95
6.3.2	LM and pronunciation dictionary . . . . .	97
6.3.3	Experiments . . . . .	98

6.4	Task-oriented lattice decoding approach . . . . .	107
6.4.1	Experiments . . . . .	108
6.5	Comparison between two proposed approaches . . . . .	113
6.6	Summary . . . . .	117
<b>7</b>	<b>Prolongation Detection System</b>	<b>119</b>
7.1	Introduction . . . . .	119
7.2	Approaches for prolongation event detection . . . . .	122
7.2.1	Supervised approach . . . . .	122
7.2.2	Unsupervised approach . . . . .	126
7.3	Experiments . . . . .	130
7.3.1	Metrics . . . . .	130
7.3.2	Supervised approach . . . . .	130
7.3.3	Unsupervised approach . . . . .	136
7.4	Summary . . . . .	137
<b>8</b>	<b>Detecting and Classifying Stuttering Events in Transcriptions</b>	<b>139</b>
8.1	Introduction . . . . .	139
8.2	Previous work . . . . .	141
8.3	Task definition . . . . .	143
8.4	Machine learning classifiers to detect stuttering event from transcription . .	144
8.4.1	Conditional random fields . . . . .	145
8.4.2	Bidirectional LSTMs . . . . .	145
8.5	Features of the classifiers used to detect stuttering events . . . . .	147
8.5.1	Word/Utterance-based features . . . . .	147
8.5.2	Character-based features . . . . .	148
8.5.3	Word embedding . . . . .	148

---

8.6	Experiments . . . . .	149
8.6.1	Data transcription and annotation . . . . .	150
8.6.2	CRF classifier: effect of adding more data . . . . .	152
8.6.3	BLSTM . . . . .	156
8.6.4	Evaluation on ASR transcripts . . . . .	157
8.6.5	Decoding the ‘Arthur the rat’ passage transcribed by ASR using task-oriented lattice approach . . . . .	162
8.7	Summary . . . . .	163
<b>9</b>	<b>Diagnosing Severity level of Stuttering via a Speech Sample</b>	<b>165</b>
9.1	Introduction . . . . .	165
9.2	Assessment . . . . .	166
9.3	Diagnosis . . . . .	168
9.4	Framework evaluation . . . . .	171
9.5	Summary . . . . .	173
<b>10</b>	<b>Conclusion</b>	<b>175</b>
10.1	Research findings . . . . .	176
10.2	In support of the thesis . . . . .	179
10.2.1	Original contributions . . . . .	181
10.3	Future work . . . . .	183
	<b>Bibliography</b>	<b>185</b>
	<b>Appendix A Interview form</b>	<b>207</b>
A.0.1	Interview form : Questionnaire for parents . . . . .	207
	<b>Appendix B Reading passages</b>	<b>211</b>
B.1	‘Arthur the rat’ passage by (Abercrombie, 1964) . . . . .	211

B.2 ‘One more Week to Easter’ passage developed in UCL lab . . . . . 212



# List of Figures

1.1	<i>Framework structure. Starts with processing a speech sample through two systems simultaneously. The first system is an ASR responsible for producing a word-level transcription and analysing it by a machine-learning classifier to detect stuttering events from the transcription, then submitting it to the diagnosis system. Simultaneously, the prolongation detector processes the speech sample to detect prolongation events and then submits them to the diagnosis system. The diagnosis system then classifies stutters into three main bands: normal disfluency, borderline or beginning stuttering, based on the types and count of stuttering events.</i>	3
3.1	<i>Standard ASR system architecture.</i>	46
3.2	<i>An HMM model that includes three emitting states and both entry and exit of non-emitting states.</i>	50
3.3	<i>Construction process of a context-dependent LM through interpolation of an out-of-domain model with in-domain changes in speech.</i>	54
3.4	<i>An example word error rate evaluation.</i>	55
4.1	<i>Orthographic transcription example provided by UCLASS (Howell et al., 2009). The red boxes have been added to highlight certain stuttering events by the author of this thesis.</i>	63

4.2	<i>Graph of different reading passages included in the dataset. Name of each reading passage is provided along with its number of recordings. The Stuttering Severity Instrument-3 (SSI-3) contains texts suitable for the age of the participants (Riley, 1994).</i> . . . . .	64
4.3	<i>This figure demonstrates part of one recording transcription process using Audacity software. Because of Audacity's simple interface, a user can accurately select the start and end points of each utterance as well as scroll right/left through the audio track.</i> . . . . .	68
4.4	<i>Transcription example for speech sample after exporting the file using Audacity software. The red boxes have been added to highlight certain stuttering events by the author.</i> . . . . .	69
4.5	<i>Word-level annotation example.</i> . . . . .	70
5.1	<i>Augmentation example.</i> . . . . .	84
5.2	<i>Sound repetition augmentation example with already exist of word repetition.</i>	84
5.3	<i>Decoding, (a) example showing the ability of the ASR to detect stuttering events after applying task-oriented lattices; (b) example showing the deletion of stuttering events using the baseline ASR.</i> . . . . .	88
5.4	<i>Corresponding FST graph for the prompt 'GARDEN WITH AN ELM TREE'. The arc <b>GARDEN_PW</b> allows part-word repetition and it could be [gar] or [den] while <b>GARDEN_S</b> is sound repetition and it could be [ga]. The go-back transition allows word/phrase repetition and revision.</i> . . . . .	89
6.1	<i>WER results of the LM built from training set built from the training set.</i> . . .	99
6.2	<i>WER improvement between the LM built from the training set and LM built from the augmented corpus with artificial stuttering data.</i> . . . . .	101
6.3	<i>Example of insertions and substitutions that affect the WER but do not affect the stuttering analysis process.</i> . . . . .	102

6.4	<i>This figure represents the improvement in the miss rate after applying LM augmentation. . . . .</i>	105
6.5	<i>This figure represents the false alarm rate increase after applying LM augmentation. . . . .</i>	106
6.6	<i>Detection error trade-off (DET) for the detection of sound/word/phrase repetitions and revisions in the development set for the task-oriented decoding approach. The optimal point is reached when the best weights are applied. .</i>	108
6.7	<i>WER improvement between the deterministic LM built from the original prompt and the LM built from the task-oriented lattices. . . . .</i>	109
6.8	<i>This figure represents the misses after applying Task-oriented lattice. . . . .</i>	111
6.9	<i>This figure represents the false alarm after applying Task-oriented lattice. .</i>	111
6.10	<i>Comparison of WER improvements between the LM augmentation approach and LM built from task-oriented lattices. . . . .</i>	113
6.11	<i>Comparison of misses and FPR improvement between the LM augmentation approach and LM built from task-oriented lattices. . . . .</i>	115
7.1	<i>Main stages of the proposed integrated system. . . . .</i>	120
7.2	<i>Extraction of low-level audio features. . . . .</i>	122
7.3	<i>Classification process. . . . .</i>	123
7.4	<i>GMM supervector concept . . . . .</i>	124
7.5	<i>An example of syllable counting method. Solid line, and dashed line are energy signal, and zero-crossing rate, respectively. Circle marks are considered in the syllable counting process. . . . .</i>	128
7.6	<i>Prolongation detection. (a) Speech sample ‘may not’ with prolongation in letter ‘n’ in word ‘not’ (b) highly similar segments detected using ACF, and (c) detected prolonged segment which is longer than threshold determined by speaking rate detector. . . . .</i>	129

7.7	<i>(a) Decoding stage without filtration process. (b) Decoding stage with correlation filter process. (c) Decoding stage with correlation and smoothing-based filtration process. . . . .</i>	133
8.1	<i>Word level annotation example. . . . .</i>	143
8.2	<i>Long Short-term Memory Cell. . . . .</i>	146
8.3	<i>(a) illustrates the effects of insertion errors on the classifier performance using NIST scoring tool. In this case due to the inserted word the followed by a correct word 'the', any perfect classifier would label both the actual and the repeated the words with the label W, which is incorrect with respect to the reference transcript, which has one 'the'. (b) illustrates the effects of deletion errors on the classifier performance using NIST scoring tool. In this case due to deleting sound repetitions 'ha', the classifiers never saw the deleted sound, and mislabelled the following word as NS, which is incorrect with respect to the reference transcript. . . . .</i>	159
9.1	<i>Ground truth diagnosis recordings sample. . . . .</i>	171
9.2	<i>Confusion matrix of the predicted severity level from proposed framework with the actual severity level diagnosed by a SLP. . . . .</i>	172

# List of Tables

2.1	<i>Risk factors.</i> . . . . .	23
2.2	<i>Attempts to identifying subtyping of stuttering (Yairi, 2007).</i> . . . . .	24
2.3	<i>Stuttering development levels/types.</i> . . . . .	25
4.1	<i>The UCLASS overview information about each release. The number of recordings of each category is provided in column number two. Age range and the mean age for each category are given in NNyNNm format, where y is the year and m is the month (Howell et al., 2009).</i> . . . . .	64
4.2	<i>Types of Stuttering.</i> . . . . .	71
4.3	<i>Interpretation of Kappa (Randolph, 2005).</i> . . . . .	72
4.4	<i>Kappa statistics and their interperatation.</i> . . . . .	73
5.1	<i>Description and frequency of each stuttering class on UCLASS, Release Two, 48 read recording samples.</i> . . . . .	81
6.1	<i>WER (%) using PF-star as a training set.</i> . . . . .	94
6.2	<i>(%) using PF-star with stuttering data as a training set.</i> . . . . .	95
6.3	<i>Summary of training and test data.</i> . . . . .	96
6.4	<i>Baseline experiment results when trained on the LM built from stuttered training data. 'n/a' means that this event was not present in the test set.</i> . . .	100

6.5	<i>Miss rate and false positive rate results after applying the augmented LM approach, 'n/a' means that this event was not present in the test set. . . . .</i>	103
6.6	<i>Average miss rate and false positive rate results. . . . .</i>	104
6.7	<i>Miss rate and false positive rate results after applying the task-oriented lattice approach. 'n/a' means that this event was not present in the test set. .</i>	110
6.8	<i>Average miss rate and false positive rate results. . . . .</i>	112
6.9	<i>Miss rate and false positive rate comparison results between the LM augmentation approach and task-oriented lattices. 'n/a' means that this event was not present in the test set. . . . .</i>	114
6.10	<i>Average miss rate and false alarm results between the LM augmentation approach and task-oriented lattices. . . . .</i>	116
7.1	<i>Results of applied KNN and SVM classifiers to detect prolongation events directly from speech signal. . . . .</i>	131
7.2	<i>Comparison of achieved results with the results in the literature. n/a means not mentioned. . . . .</i>	132
7.3	<i>Results after applying GMM-supervector approach in three different cases during decoding stage. The first case demonstrate the results without applying any filters. The second results after applying a correlation-based filter. The third results after applying a correlation and smoothing filters. . . . .</i>	134
8.1	<i>Types of stuttering. . . . .</i>	150
8.2	<i>Statistical data for the training sets. . . . .</i>	152
8.3	<i>Statistical data for the test sets from the human transcripts. . . . .</i>	152
8.4	<i>CRF<sub>ngram</sub> results trained on different tasks, using ngram features only. CRF<sub>ngram</sub> trained on Task<sub>1</sub> which only using the (read) data, then on Task<sub>2</sub> which using the (read+Spon) data, then on Task<sub>3</sub> which using the (read+Spon+Art) data. . . . .</i>	153

8.5	<i>CRF<sub>aux</sub> results trained on different tasks, using auxiliary features. CRF<sub>aux</sub> trained on Task<sub>1</sub> which only using the (read) data, then on Task<sub>2</sub> which using the (read+Spon) data, then on Task<sub>3</sub> which using the (read+Spon+Art) data.</i>	155
8.6	<i>BLSTM results trained on different tasks, using embedded features. BLSTM trained on Task<sub>1</sub> which only using the (read) data, then on Task<sub>2</sub> which using the (read+Spon) data, then on Task<sub>3</sub> which using the (read+Spon+Art) data.</i>	156
8.7	<i>Summary table for results on human transcript, using classifiers trained on Task<sub>3</sub>.</i>	157
8.8	<i>Results of classifiers trained on Task<sub>3</sub>, on ASR transcripts (against ASR_scoring).</i>	160
8.9	<i>Results of classifiers trained on Task<sub>3</sub>, on ASR transcripts (referens<sub>1</sub>).</i>	161
8.10	<i>Results of CRF<sub>aux</sub> trained on Task<sub>3</sub> on ASR transcripts for children who read the ‘Arthur the rat’ passage against “Human_scoring”, which is the actual labelled human transcript using the annotation approach proposed by Yairi and Ambrose (2005).</i>	163
9.1	<i>This is an example of the approach based on counting disfluencies for sample containing 229 words.</i>	168
9.2	<i>Stuttering severity levels.</i>	170





# Chapter 1

## Introduction

Imagine listening to a child who stutters, trying to parse each word and transcribe it. A speech language therapist usually does this tedious task offline after recording the clinical session, as it is often impossible to do so in real time during the session. This process takes a long time, for every spoken word must be transcribed, which takes time, effort and requires knowledge of the relevant categories.

In this thesis, we aim to build an automatic framework to aid therapists in diagnosing children who stutter by producing an automatic transcription that includes stuttering events using automatic speech recognition (ASR), analysing the transcription by detecting and classifying each events in the transcription, and evaluating the severity level of stuttering in a speech sample. To do so, we will adapt the ASR system to recognise different stuttering events.

### 1.1 Motivation

Stuttering is a speech disfluency disorder that typically begins in childhood. It usually manifests by age four in 95% of sufferers (Yairi and Ambrose, 2013), just as a child is learning to talk. It is a complex and uncontrolled disorder that can cause a wide range of

social and mental problems (Iverach et al., 2009; Tran et al., 2011). Inadequate diagnoses and intervention at an early age can increase the risk of the condition becoming chronic and can have negative consequences on children with stuttering and their families (Hayhow et al., 2002; Craig and Calver, 1991). It is therefore critical to address speech disorder problems in early childhood with proper, appropriate medical intervention (Iverach et al., 2009; Tran et al., 2011). The prognosis for full recovery dramatically lessens if the stuttering persists into adolescence. The risks increase during adulthood rather than childhood because most young children are not fully aware of their disfluency. Furthermore, it is not possible to determine a child's chance of naturally recovering, and children are less tractable as they get older due to a reduction of neural plasticity (Jones et al., 2005). Therefore, clinician intervention should take place as early as the preschool years.

During the assessment phase, clinicians need to carefully measure the stuttering events to determine the severity of the stuttering. This measurement is usually conducted by counting the number of stuttering events in the child's speech; however, this process is extremely dependent on the clinician's experience (Brundage et al., 2006). Another approach, which provides more details about improvement and progress for the child's case, has the clinician transcribing a recorded session and classifying each spoken term into one of several categories, including normal, disfluent or stuttering (Gregory et al., 2003). This process is considered a difficult task because it is time consuming and requires requisite knowledge of each stuttering event. An automated speech transcription of the recorded speech using ASR could help clinicians speed up the assessment process and store the data for further investigations.

To realise this goal, the ASR system must be adapted to recognise full verbatim utterances, including pseudo-words and non-meaningful part-words. This study therefore contributes to the limited available data for stuttering speech and proposes new approaches to address this

problem while preserving a full verbatim output of stuttering speech. Finally, this information will be analysed to determine the severity level of stuttering.

## 1.2 Research focus

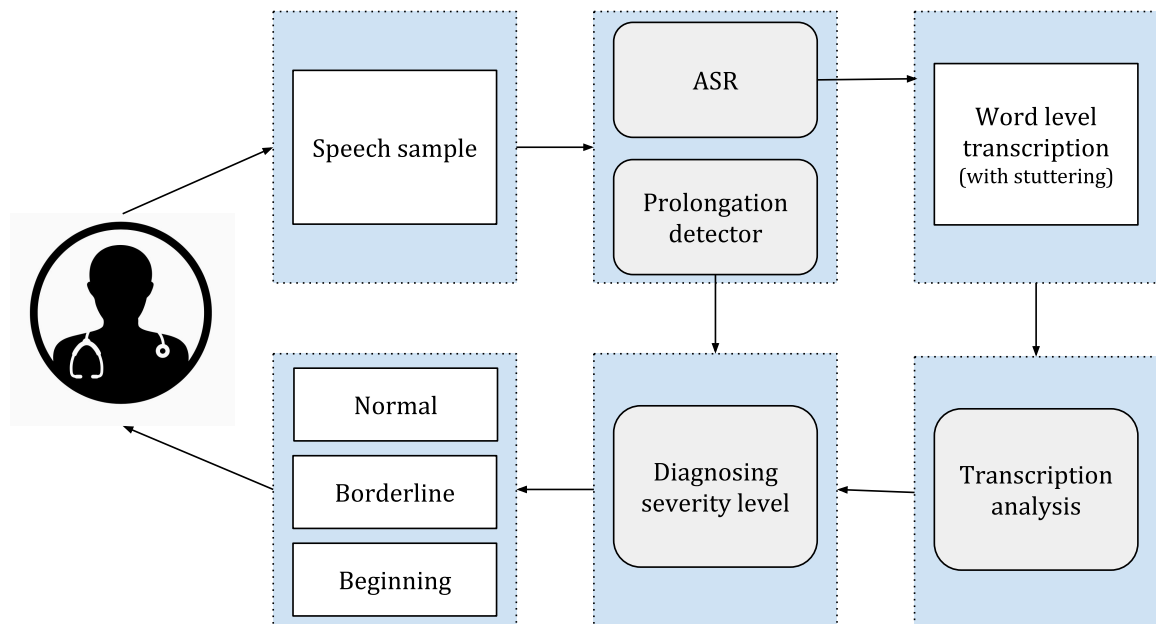


Figure 1.1 *Framework structure.* Starts with processing a speech sample through two systems simultaneously. The first system is an ASR responsible for producing a word-level transcription and analysing it by a machine-learning classifier to detect stuttering events from the transcription, then submitting it to the diagnosis system. Simultaneously, the prolongation detector processes the speech sample to detect prolongation events and then submits them to the diagnosis system. The diagnosis system then classifies stutters into three main bands: normal disfluency, borderline or beginning stuttering, based on the types and count of stuttering events.

Increased attention has been paid to advancing clinical applications using speech recognition and machine learning techniques. However, few studies have been undertaken to explore the value of integrating both fields to build a beneficial diagnostic clinical application for a speech disorder such as a stuttering (Geetha et al., 2000). As shown in Figure 1.1, this thesis demonstrates how the different research fields can benefit from each other by using an

adapted speech recogniser to address the problem of providing an automated full-verbatim transcription of a speech sample and analysing this using a machine-learning technique to provide a final indication of the severity level of stuttering. This task could be considered a crucial step in helping therapists use different technologies to save time and efforts.

Diagnosing the severity level of stuttering via a speech sample is considered an intriguing task that requires the integration of speech recognition fundamentals to generate a detailed transcription for a child's stuttering speech, machine-learning techniques to analyse that transcription and the detection of a duration-based stuttering class, such as prolongation.

Some studies in the literature have tried to create an automated tool that classifies rather than detects some stuttering events directly from a speech signal using different machine-learning approaches to distinguish between fluent and stuttered speech segments (Howell et al., 1997; Chee et al., 2009; Świetlicka et al., 2013). However, the paucity of training data makes the task challenging to propose direct stuttering event detection approaches, which require much more data to detect recurrent relationships in continuous speech for diagnosis purposes.

Moreover, little work has been done to explore ASR for recognising stuttering events (Nöth et al., 2000; Heeman et al., 2011, 2016). Although studies have started to explore the adaption of ASR systems in performing the task of recognising stuttering speech, the main objective of obtaining an automated transcription produced from the ASR was not achieved. The main research question from conducting such task, then, is *can we reliably diagnose the severity of a child's stuttering by automatic means?*

The first obstacle is the general scarcity of data for children, even more the scarcity of available data for stuttering children. The second concern is how to determine the best approach to adapt speech recognition technology to recognise enough stuttering events to diagnose the severity level.

To achieve this aim and answer the main research question, we have developed a four-stage approach, involving a combination of ASR, to exploit frequency-information and autocorrelation system to exploit time-information, with a classifier to analyse the transcript of the ASR output, and, finally, using that classifier to combine and analyse this information to perform the final diagnosis.

For the first stage, the ASR is conducted using two different approaches. The first approach is a conventional ASR trained on the PF-Star corpus (Russell, 2006) and UCL Archive of Stuttered Speech (UCLASS) data (Howell et al., 2009), but with the language model artificially augmented with pseudo-words and repetitions to increase the frequency of stuttering events (sound, part-word and whole-word; phrase repetition and revision). The second approach is a task-specific lattices developed from the original reading prompts, with inserted hand-tuned transitions for stuttering events. This stage will answer the research question of: *Given that there is scarce stuttering data available to train an ASR, which is the best ASR approach for detecting stuttering events?*

However, prolongation events are time-dependent, and so are detected independently using frame autocorrelation weighted by speaking-rate, which we developed into an independent detector that worked in tandem with the ASR system. This stage will answer the research question of: *How do we reliably detect time-based stuttering events when we do not have enough data, and what would the best techniques be?*

The next stage deals with the analysis of the ASR's output, which is a transcription. Each transcript is analysed using two machine learning approaches —conditional random fields (CRF) and bi-directional long-short-term memory (BLSTM)— to detect specific types of stuttering events (including sound, part-word, word and phrase repetitions). This stage will answer the research question of: *What is the best approach for detecting stuttering events from transcription?*

All stuttering events are then submitted to a classifier inspired by Guitar's (2014) diagnosis method, which classifies stutterers into three main severity levels: normal disfluency, borderline or beginning stuttering, based on the types and count of stuttering events.

### 1.3 Major contributions

The research described in this thesis provides original contributions to the field of clinical applications, applying automatic speech recognition technology to children's read speech. The major contributions can be summarised as follows:

#### **Contribution 1: Corpus Generation and Analysis**

The initial phase to achieve our goal of building an automatic diagnostic framework of stuttered speech was creating a proper dataset that could be used for development and evaluation of the ASR. The only publicly available resource is UCLASS, provided by Howell et al. (2009). However, no transcriptions or annotations are provided for the recording samples. Therefore, generating a time-aligned word level transcription including all stuttering events for every read sample of UCLASS Release Two helped to build a proper adapted ASR and utilise this resource as a ground truth for evaluation. We also created a word-level annotation to train different classifiers to detect stuttering events from ASR transcriptions.

#### **Contribution 2: Adapted ASR system to detect stuttering events using two different approaches**

1. The first approach is based on building an ASR that focused on a small stuttered corpus within UCLASS (Howell et al., 2009) and a larger corpus of clean child speech provided by the PF-Star dataset (Russell, 2006), but with the language model artificially augmented with pseudo-words and repetitions to increase and detect the frequency of stuttering events, including sound, part-word and whole-word repetition, as well as

- phrase repetition and revision (published and presented in WOCCI 2017, (Alharbi, Simons, Brumfitt and Green, 2017))
2. The second approach is a task-specific lattices developed from the original reading prompts to build a more constrained and specified language model (LM) that allowed for a number of stuttering events (published and presented in Interspeech 2018, (Alharbi et al., 2018))

### **Contribution 3: Adaptive prolongation detector**

This system focused on the prolongation events which are time-dependent. Prolongation is the uncontrolled extension of both vocalized and non-vocalized sounds. This system proposes a refined workflow to preserve time and count information, detecting rather than classifying a prolongation event. To address this problem, the prolongation events are detected independently using frame autocorrelation weighted by speaking-rate (published and presented in Interspeech 2018, (Alharbi et al., 2018))

### **Contribution 4: Analysing transcripts produced by the ASR using machine learning classifiers**

Next is the evaluation of two machine learning approaches' ability to detect stuttering events from both human and ASR transcripts of children's read speech and compare the performance between them. This comparison is conducted using lexical, and contextual features. In addition, a study was conducted on the effect of adding more data to the performance of the classifiers. This study also investigated the effect of augmenting the available training data with artificially generated training data to improve the classifiers' performances. Finally, this work described a method for studying the effect of ASR errors on classifiers' performances (published and presented in International Conference on Statistical Language and Speech Processing 2017, (Alharbi, Hasan, Simons, Brumfitt and Green, 2017))

**Contribution 5: Diagnose severity level of stuttering of a speech sample**

All detected stuttering events were submitted to a classifier based on Guitar (2014) diagnosis method, which classifies stutters into three main severity level based on the types and count of stuttering events: normal disfluency, borderline and beginning stuttering.

## 1.4 Thesis overview

The remainder of this thesis is organised as follows:

1. **Chapter 2: What Is Stuttering?.** This chapter evaluates numerous studies to illustrate the nature of the stuttering disorder. Subsequently, the most useful assessment approaches to diagnose the stuttering disorder, different treatment methods and several opinions regarding the stuttering treatment field are also presented.
2. **Chapter 3: Automatic Stuttering Recognition Background.** This chapter reviews work relating to recognition systems of stuttering speech disorder, as well as previous research that has presented various methods and algorithms for classifying stuttering events from speech signals.
3. **Chapter 4: Word level Transcription and Annotation of Stuttering Data.** This chapter contains the description of the UCLASS corpora, which is used for the experiments in the following chapters. This chapter presents the initial stage of the proposed framework, which is the creation a proper transcriptions for the UCLASS corpus to build and evaluate an ASR system. The generated corpus contains the transcription process of reading speech recordings for UCLASS Release Two. Then, it describes the annotation process of each word, to train the automatic classifier that automatically detects different stuttering events on ASR transcriptions using state-of-the-art machine-learning algorithms.



4. **Chapter 5: Automatic Stuttering Recognition.** This chapter presents two ASR approaches for the automatic recognition of stuttered speech in children. The first uses an augmented LM approach, and the second is a task-oriented lattice. The crucial advantage of ASR-based approaches is that they provide textual information.
5. **Chapter 6: Experimental Results of ASR Proposed Approaches.** This chapter evaluates the performance of the two proposed ASR approaches explained in Chapter 5, detecting sound, word, part-word and phrase repetitions, as well as revisions and interjections. Moreover, it demonstrates, discusses and compares the results of the two previous techniques.
6. **Chapter 7: Prolongation Detection System.** This chapter demonstrates and compares two proposed methods for detecting prolongation events directly from a speech signal. The first of these is a supervised approach and involves three supervised machine learning techniques. The second method uses an unsupervised approach to detect prolongation segments without any need to train a classifier.
7. **Chapter 8: Detecting and Classifying Stuttering Events in Transcriptions.** This chapter evaluates the ability of two machine learning approaches —conditional random fields (CRF) and bi-directional long-short-term memory (BLSTM)— to detect stuttering events from both human and ASR transcripts of children’s read speech and compares their performance. This comparison is conducted using lexical, and contextual features. In addition, this chapter studies the effect of adding more data to improve the classifiers’ performances. This chapter also investigates the effect of augmenting the available training data with artificially generated data to improve the classifiers’ performance. Finally, a method for studying the effect of ASR errors on classifiers’ performance is described.

8. **Chapter 9: Diagnosing Severity level of Stuttering via a Speech Sample.** This chapter evaluates the ability of our proposed framework to analyse each recording sample and provide an indication of the severity level of stuttering from an acoustic point of view. This evaluation is conducted by comparing our predicted framework severity level results with the results reported by two UK registered SLPs of the same recording samples.
9. **Chapter 10: Conclusion.** This chapter concludes the work proposed in this thesis and provides some suggestions for future research directions.

## 1.5 Published work

The list of publications obtained while this doctoral work was done and the expected publications are:

1. Detecting Stuttering Events in Transcripts of Children's Speech. *Sadeen Alharbi, Madina Hasan, Anthony-JH Simons, Shelagh Brumfitt, and Phil Green.* International Conference on Statistical Language and Speech Processing. Springer, Cham, 2017.
2. Automatic recognition of children's read speech for stuttering application. *Sadeen Alharbi, Anthony-JH Simons, Shelagh Brumfitt, and Phil Green.* Proc. WOCCI 2017: 6th International Workshop on Child Computer Interaction. 2017.
3. A Lightly Supervised Approach to Detect Stuttering in Children's Speech. *Sadeen Alharbi, Anthony-JH Simons, Shelagh Brumfitt, and Phil Green.* Proc. Interspeech 2018.
4. Sequence Labeling to Detect Stuttering Events in Read Speech. *Sadeen Alharbi, Anthony-JH Simons, Shelagh Brumfitt, and Phil Green.* Computer Speech & Language (2018): (submitted).

- 
5. Automatic Framework to Aid Therapists to Diagnose Children who Stutter. *Sadeen Alharbi, Anthony-JH Simons, Shelagh Brumfitt, and Phil Green*. *Biomedical Signal Processing and Control* (2018): (In preparation).



# Chapter 2

## What Is Stuttering?

### 2.1 Introduction

Human beings communicate in variety of rich forms through writing, speaking and other non-verbal communication such as eye contact and hand gestures. The most powerful and important method of human communication is speech. One component of speech effectiveness relies on fluency. Fluency involves effortless flow of speech. It also involves pausing, rhythm, intonation, stress, rate, information flow, speed, and effort. Any disruption in fluency is called disfluency, which includes several forms. Stuttering is one common speech disfluency disorder. In the late 1950s, Johnson proposed the first classification of different types of this speech disfluencies, which include sound/part/word and phrase repetition, prolongation, revision, interjection, broken word and incomplete phrase (Johnson, 1961). Ever since, this classification of disfluent speech has been used by clinicians and researchers.

In this chapter, we will be investigating numerous studies to illustrate the nature of the stuttering disorder. Subsequently, the most useful assessment approaches to diagnose the stuttering disorder will be explained. Varying treatment methods will be demonstrated with

several opinions regarding the stuttering treatment field. Finally, we provide a brief summary to encapsulate the most important points in this chapter.

## 2.2 Nature of stuttering

### 2.2.1 Overview

The terms ‘stuttering’ and ‘stammering’ are used interchangeably. A stuttering disorder manifests as an abnormally high number of stoppages in the forward flow of speech. Definitions of stuttering outline common speech symptoms and other features that may help to determine a proper diagnosis of it, although there is still no clear method for determining whether a child stutters or not (Howell, 2011). Yairi and Ambrose (2005) studied children who exhibited certain interruptions in speech flow - such as repeating sounds, part words or whole words with prolongations and blocks - and consequently categorised these speech phenomena as stuttering-like disfluencies (SLD). Succinctly, the World Health Organisation defines stuttering as follows,

“disorders of rhythm of speech in which the individual knows precisely what he wishes to say, but at a times is unable to say it because of involuntary, repetitive prolongation or cessation of a sound”, (WHO, 1957, pg. 202)

This disorder usually develops as children create sentences and combine words, and it is associated with negative social interactions in children, teenagers, and adults. Because stuttering usually begins in childhood, most research on stuttering is related to children. Yairi and Ambrose (2013) assert that 95% of children who stutter begin to stutter when they are four years of age or younger and that nearly all begin to stutter before they are 12 years of age. In addition, other studies have shown that stuttering generally begins in children between two and six years of age and that many children who start to stutter recover from

the disorder with only a limited amount of therapy (Shimada et al., 2018). Typically, most clinicians call this situation a spontaneous recovery or natural remission (Andrews et al., 1983; Bloodstein, 1969; Jiang et al., 2012; Shimada et al., 2018). Notably, if an individual has not recovered after his or her teenage years, the chances of a full recovery are dramatically reduced. Persistent developmental stuttering will continue in only 20% of children who stutter (Jiang et al., 2012). The following section demonstrates the impact of stuttering.

### **2.2.2 Impact of stuttering**

Stuttering is an uncontrolled disfluency disorder. As a complex disorder, it can cause a wide range of mental and social problems (Iverach et al., 2009; Tran et al., 2011). Crucially, this disorder might become permanent if a child does not receive proper treatment for it (Bloodstein, 1969). The increasing incidence of stuttering has been reported as 8.5% for preschool children (Reilly et al., 2009). Stuttered children may obtain negative responses from their fluent peers, which can negatively affect their social interactions (Langevin, 2009). Furthermore, these children are more likely to experience developed levels of frustration and social withdrawal than their fluent peers (Langevin et al., 2010). These negative effects can continue into adolescence, as many teenagers who stutter experience bullying (Blood and Blood, 2007). Many have difficulty building friendships later in life because of bullying during their school years (Mooney and Smith, 1995). This disorder is considered a chronic condition during adulthood because of its associated negative effects, such as quality-of-life disruptions (Craig et al., 2009), communication difficulties (Craig, 1998; Craig and Tran, 2006), and job-performance issues (Klein and Hood, 2004). In addition, mental health issues increasingly occur in adults who stutter, such as social anxiety (Blumgart et al., 2010). These issues usually manifest during adulthood rather than childhood because most small children are not entirely aware of their disfluency. The following sections describe how stuttering can influence different aspects of a person's life.

- **Communication difficulties**

Studies reported notable variations in self-perception communication anxiety (Blood et al., 2001) and self-perception communication ability between a teenager who stutters and their fluent peers (Blood and Blood, 2004; Blood et al., 2001). One of the major obstacles to developing communication ability and skills is the high levels of communication anxiety between adults who stutter. Blood et al. (2001) linked the lack of self-perception of communication ability to the reduction in interactions, and social withdrawal. It would be reasonable to consider that the failures of communication for all these years could lead teenagers to develop these negative communication attitudes.

- **Impact on families**

Families have a vital influence on their children's life, and they are often involved in the treatment process of their child's stuttering. However, there is a limitation in the studies regarding the impact of stuttering on families. Hearne et al. (2008) mentioned in his study that two teenagers claimed that their parents were helpful, but they often tried to avoid talking about stuttering. Also, there were some adults in the study who stuttered and mentioned that the stuttering had a positive impact on the relationship between them and their parents. (Klompas and Ross, 2004). On the other hand, around one in five in the Hearne et al. (2008) study claimed that their parents showed an impatience and misunderstanding. Furthermore, there were other participants who claimed that stuttering had been a hard situation for their parents to handle (Klompas and Ross, 2004). In short, there have been diverse opinions shown and different experiences seen regarding how sufferers and their families deal with the stuttering; generally, some people experience patience and understanding while others report that they experience a variety of different family issues as a result of their disorder (Klompas and Ross, 2004).



The negative influence of stuttering has also been extensively mentioned in early childhood literature (Cook and Howell, 2013; Langevin, 2009; Yairi and Ambrose, 2005). There is a risk for children who stutter to be bullied and as aforementioned, stuttering often has a significant psychosocial effect on child sufferers. It is important to realize the impact that an increased risk of suffering from bullying and general psychological effects can have on a child's life (Cook and Howell, 2013).

### 2.2.3 Etiology

From a therapist's point of view, one of the major questions that could be asked by parents is what are the causes of stuttering? While unfortunately there is no absolute cause for the disorder, parents can find some consolation as experts have found four main clarifications about the causes of stuttering (Yairi and Ambrose, 2005). Speech therapists refer to the possible influence of stuttering as a clarifications rather than causes:

- **Psychogenic clarifications**

This clarification indicates that the stuttering is a symptom of another serious problem. Sometimes when there are emotional difficulties during childhood, the child faces problems handling these emotional conflicts. Then, these conflicts may manifest as speech difficulties. However, many people consider stuttering as a definite psychological problem which is an unacceptable assumption among many speech therapists today. There is no absolute evidence that psychogenic problems in children are the cause of stuttering when they are compared with non-stuttering children. Different kinds of psychological problems can be caused by suffering from stuttering rather than be the cause of it, particularly, when the stuttering is persistent.

- **Learning clarifications**

The intent of ‘learning’ in childhood is to develop and make a habit out of certain favorable manners or routines. Yairi and Ambrose’s book explains this type of theory by providing an example (Yairi and Ambrose, 2005). There are some normal repetitions when a child begins to talk such as ‘Da Da Daddy’. If the parents are always trying to correct the child’s pronunciation, then the child may become more sensitive to these repetitions. In this case, due to the child’s self-consciousness about their stuttering, this may increase the repetitions rather than be corrected. This clarification is problematic however as it places all of the blame on the parents for their child’s stutter. For this reason the theory is not accepted by most experts in the field.

- **Organic clarifications**

This explanation finds the reason for stuttering to stem from brain problems, linking a child’s difficulty in coordinating movements of speech muscles to stuttering. A long time ago several reasons were held up as organic causes of stuttering, such as the stutterer having a long tongue or having been forced to write right-handed when they are left-handed. However, there remains no method to identify a particular abnormality. Recent research does indicate brain differences between the person with stuttering and fluent peer; although, no research has been conducted on children who stutter.

- **Complex clarifications**

This explanation is a combination of all previous clarifications with other concepts. Here, the experts are sure that stuttering could be transmitted genetically. However, the researchers are not sure about what exactly is transmitted. The person with stuttering might inherit an organic element or a certain function which may increase their likelihood of becoming a stutterer. In addition, there are many studies that point

to the role of the environment, as there is a strong interaction between the environment and genetic factors.

#### **2.2.4 Disfluency characteristics of early stuttering**

This section is written as a summary of some research and perspectives on vital factors in stuttering. Most of the main areas are covered in this section. Starting with a description of heredity that is provided as one of several possible background factors which can contribute to stuttering. This factor can affect the brain structure and development functions. Furthermore, the difference in the brain structure could have a major impact on motor control. Also, there are two more factors which appear to influence the likelihood of an onset of stuttering and its subsequent development: language factors and temperament factors.

- **Hereditary factors**

There are two interesting methods to study the relation between heredity and stuttering: family studies and twins studies. Researchers have used these two approaches to prove that stuttering can be transmitted genetically (Yairi and Ambrose, 2005; Guitar, 2014).

*Family Studies:* Family studies have supplied strong evidence for a genetic predisposition in many people who stutter (Yairi and Ambrose, 2005; Howell, 2011; Guitar, 2014). By conducting several studies on different groups of people who stutter and comparing them with other groups of people who do not stutter, it has become clear that stuttering can run through families and can be transmitted genetically (Yairi and Ambrose, 2005). From clinicians' point of view, informing the parents that the stuttering is usually inherited rather than as a result of their parenting is a significant advantage.

*Twin Studies:* Speech and language pathologist (SLP) refer to pairs of twins when they

both stutter as concordance. Studies on twins have proven that concordance among identical twins is higher than concordance among fraternal twins. However, even though there is much concordance between identical twins, there are many identical twins who in which one twin stutters while the other does not - a phenomenon known as discordance. Discordance between identical twins suggests that the environment must also be a contributory factor with genes to produce stuttering disorder (Guitar, 2014).

*Brain Function and Structure:* According to Guitar (2014), the brain structure and functions in the central nervous system for people who stutter would be different from normal disfluent peers. The structure and function of an individual's brain could be affected by a genetic predisposition, trauma, injury, or unknown cause which leads to a delay in normal neural processing for language and speech in general. This disruption may result in the different kinds of reparations, such as revision, word or part-word repetitions, and blocks, these symptoms are only seen in stuttering. Brain damage can also cause difficulties with language production, for example, saying the names of objects and grammatical construction. In summary, heredity should not be considered to be the only cause of stuttering; rather, clinicians should also focus on and ask the parents about the events near the time of the onset of stuttering.

- **Language factors**

Yairi and Ambrose (2005) discuss the domain of linguistic elements and language progress in children with stuttering. From a therapist's point of view, the results of the experiments conducted by the University of Illinois Stuttering Research Program do not reveal a clear direction. For example, the findings disclosed that stuttering might be the cause of lower achievement in language development. Another experiment hypothesis that the complexity of the grammar and the words' position might effect the stuttering event. Thus, it is evident that the linguistic variables in general have a substantial

effect on a child who stutters. However, the influence of language on stuttering can also be illustrated by language development, language delays and language complexity (Guitar, 2014):

*Language Development Factor:* There is a clear impact of the language development on stuttering because the language development in a child puts a stress on his speech production.

*Language Delay/ Disorder Factor:* This factor might quicken a child's stuttering problem or even make it worse because such a child has two main deficits: a speech motor control problem and a language problem. Therefore, children who stutter and their families might turn the foci of their resources and attention away from the speech motor control problem and towards the language problem. Many studies have illustrated language delays and difficulties in children who stutter (Ntourou et al., 2011). They have suggested that the overall language, receptive vocabulary, expressive vocabulary and the mean length of utterances may cause a disruption in the fluent initiation and/or continuation of speech-language, which is most typically characterised by the production of speech disfluency.

*Language Complexity Factor:* This factor deals with the sensory-motor control of speech and how it may trigger the occurrence of stuttering. Different studies have shown that when the person who stutters says a long or linguistically complex sentence, they are more likely to stutter. Also, there are other different utterance positions which can increase the possibility of stuttering such as grammatical word types, longer words and words at the beginning of an utterance. In summary, it is important during the evaluating phase of a child who has recently started to stutter to determine whether the child is/was in a period of intense language development when the stuttering started. Also, stuttering might be reduced in children who are beginning to stutter when there is

a reduction of the linguistic load. This reduction can be achieved by the use of pauses and a slower rate of speech.

- **Temperament factors**

Many of the clinicians who work with children who stutter have reported that parents often describe their children as sensitive. The same parents also mentioned that their children were sensitive even before stuttering begin (Jones et al., 2017). For example, the child easily gets upset when something changes in their routine life, and the child can be very shy when meeting some strangers. These emotional characteristics might be considered a part of the child's temperament. Temperament is defined as the contribution of the child's biological features to their own emotional, cognitive, and motor characterization. So, temperament is the emotional reactivity the child is born with. There are a number of authors who considered studies on a reactive temperament to enhance the understanding of the nature of stuttering e.g. (Peters and Guitar, 1991; Walden et al., 2012). The reactive temperament may increase the physical tension in a child when they are disfluent normally, which leads to the creation of a long cycle of disfluency before finally resulting in the development of a stutter.

### **2.2.5 Risk factors**

The point that there are different factors that place a child at increased risk of developing the stuttering is related to the lack of a particular cause to this disorder (Ward, 2017). Risk factors are the elements within the child or within the environment of the child which can increase or decrease the persistence of their stuttering, which can also influence the appropriate level of treatment. Table 2.1 describes a summary of some factors.

Table 2.1 *Risk factors.*

Factors Relating to the Child	Factors Relating to the Environment
<b>Age.</b> Studies shows that preschool children are at risk of developing a stutter (Ward, 2017). Also, most children who stutter have started between the age of 2 and 6, and almost all stuttering cases begins before age 12 (Jiang et al., 2012).	<b>Family communication style.</b> Different studies have mentioned the relationship between the language complexity of the parents and persistent stuttering of the child (Kloth et al., 2000).
<b>Family History.</b> A child who has one of their parents or relatives who suffering from persistent stuttering and recovering with treatment is more likely to suffer from persistent stuttering as well (Ambrose et al., 1997).	<b>Family Expectations.</b> High expectations might increase the tension of a child as a negative reaction, leading to the possibility of development of a persistence stutter (Guitar, 2014; Ward, 2017).
<b>Gender.</b> Persistent stuttering is more likely to happen in boys than in girls which is about 2:1 in young children (Ambrose et al., 1997).	<b>Life events.</b> Emotion conflicts and events during the child's life will affect the child's stuttering (Guitar, 2014).
<b>Speech and language skills.</b> One study mentioned that the variation between words and syntax might indicate that the child is at risk of long-term stuttering (Conture, 2001).	<b>Family's schedule.</b> The stressful and less predictable schedules of many wealthy families can cause tension in children, leading them to develop a stutter (Guitar, 2014).
<b>Sensitivity/temperament.</b> Children seen to be sensitive may need longer treatment (Richels and Conture, 2010).	<b>Others' reaction to stuttering.</b> Family reaction is very important to the child who suffers from stuttering. An impatient family can put stress on their child, aggravating their condition(Guitar, 2014).
<b>Reactions to Stuttering.</b> Treatment may be needed for children who suffer poor self-esteem as a result of their stutter or who feel self-conscious.	

### 2.2.6 Identifying types of stuttering

- **Background on identifying types of stuttering** This section mainly aims to summarize a wide-ranging review of various attempts at subtyping stuttering during the past 50 years. This review conducted by (Yairi, 2007) and it is based on subtyping the stuttering according to seven categories which reflect the different authors' methods. The categories include: general etiology, reactions to drugs, prominent stuttering phenomena, concomitant disorders, the developmental course, biological characteristics, and statistically-generated models. Table 2.2 illustrates the summary of all attempts of subtyping stuttering by (Yairi, 2007).

Table 2.2 *Attempts to identifying subtyping of stuttering (Yairi, 2007).*

Category	Basis	Sub Categories	Examples
General Etiology	Attempts to subtype stuttering based on the general causes of the disorder.	Single Domain	Brill (1923), Canter (1971)
	However, some approaches focus on Single etiology domain while others view subtypes as representing mixed etiology domains.	Mixed Domains	Van Riper (1947), St. Onge (1963), Luchsinger and Arnold (1965)
Prominent stuttering phenomena	Mode of Expression	Exteriorized-interiorized	Douglas and Quarrington (1952)
	Type of Disfluency	Clonic-tonic	Froeschels (1943)
	Adaptation effect	Reading-conversation	Newman (1963)
	Stuttering severity	Mild-Moderate-severe	Watson and Alfonso (1987)
Reactions to drugs	Only one classification related to neuropathologies		
Biological characteristics	Gender	Male-female	Silverman and Zimmer (1979, 1982)
	Hemispheric lateralization	Ear preference	Hinkel (1971)
	Handedness	Ear preference and sex	Foundas et al. (2004)
	Brain morphology	Responses to DAF	Foundas, Bullich, et al. (2004)
	Genetics	Positive- negative	Seider et al. (1983).
Concomitant disorders	Language/motor		Riley (1971)
	Presence/absence		Blood and Seider (1981)
	Neurological and ADD		Alm and Risberg (2007)
Developmental course	Age at onset	Three age levels	Dostalova and Dosuzkov (1966)
	Developmental diversity	Four tracks	Van Riper (1971)
	Developmental diversity	Recovery/treatment	Conture (1990)
	Persistency-recovery	Genetic loading	Ambrose et al. (1997)
Statistically-generated models	Audible-visible		Prins and Lohr (1972)
	Multiple factors		Andrews and Harris (1964)
	Symptom-based		schwartz and Conture (1988)
	Diagnostic component model		Riley and Riley (2000)



- **Subtypes of stuttering (Stuttering development)** Guitar (2014) shows a model of stuttering development and how it should be treated/handled differently at each level. This model is a sequential model from five stages (types): the first stage is simply a normal disfluency. Then, the following three stages - borderline stuttering, beginning stuttering, and advanced stuttering- indicate the growing levels of the stuttering development. This thesis will focus on only the first three types as we can obtain an indication of these levels by analyzing the speech samples and by calculating the core behaviours. Behaviours such as blocks and stopping air flow are not considered at the moment. Table 2.3 explains each level/type and demonstrate how to distinguish each type.

Table 2.3 *Stuttering development levels/types.*

Category	Characteristics
Normal disfluency	No greater than 10 disfluencies per 100 words. One-unit repetitions, sometimes two. Most common disfluency types are interjections, revisions, and word repetitions.
Borderline stutter	More than 10 disfluencies events per 100 words. More than two-unit repetitions. Disfluencies relaxed and loose. Child mostly not aware of his disfluencies.
Beginning stutter	Muscles reveal tension and hurry appear in stuttering. More continued prolongations and repetitions. Repetitions are rapid and irregular. Awareness of difficulty and feeling of frustration are present.
Advanced stutter (persistent)	Trying to avoid stuttering which leads to very rapid and strong stuttering. Most frequent core behaviours are longer such as a blocks. Emotion of fear and embarrassment are very strong. Stutterer has a deep negative feeling about themselves as a person.

## **2.3 Assessment and diagnosis of stuttering**

Usually, most children's parents correctly note signs of speech disfluency and they can differentiate the stuttering from other disorder. The first assessment, as described in the following section, starts with obtaining extensive information about family history, gathered through a parent interview, which emphasizes different aspects of the child's onset of stuttering. After that, more critical assessments should be done to confirm the level of stuttering. This includes specific practical assessment approaches to aid the therapist in understanding a client and their stuttering problem. However, therapists disagree on how best to assess the stuttering. The following section will describe how to cover the background and case history. The next section will demonstrate the assessment approach provided by Guitar (2014) and explain more about the disagreement on assessment.

### **2.3.1 The need for background and case history**

The assessment process should cover the standard items, such as personal identity, family background, health, and physical and speech development; such information should then be written in the case history. The speech and language pathologist (SLP) should also ask additional questions that aims to clarify the events taking place at the onset of the stuttering up to the current time. Also, this questioning process is very important in order to obtain a general idea about the progress of the disorder, as this is considered a key element in the overall assessment (Yairi and Ambrose, 2005; Guitar, 2014). Furthermore, evaluating all the information obtained from the interview against the previous risk factors (described in section 2.2.5) will help the clinician determine the final judgment. As an example, the following questions are provided by Yairi and Ambrose (2005):

- When did the child start stuttering?
- What were the initial signs or characteristics?
- What were the circumstances surrounding onset?
- How has the stuttering progressed?

Other relevant questions could include

- When did the child start talking?
- Can you understand your child's speech?

### **2.3.2 Assessment approach**

This section will present in more detail the approach provided by Guitar (2014), not because it is the only assessment approach but because it is the most recent one and it is a comprehensive approach and takes into account many of the findings of the previous people as well.

This assessment approach has different phases which include: parent-child-interaction, clinician-child-interaction and speech sample. The speech sample should contain 200 syllables or more to be more accurate.

The SLP can decide whether the child is suffering from stuttering or not by analyzing the different variables in phase known as 'determine the pattern of disfluencies':

1. Frequency of disfluencies: This is computed by extracting the number of disfluencies per 100 words.
2. Disfluencies types: There are 8 types of disfluencies that include part word repetition (PW), sound repetition, multi syllable word repetitions, phrase repetition, interjections, revisions (incomplete phrase), prolongations and tense pauses. This

variable can be expressed by calculating the percentage of total disfluencies which are under the stutter-like disfluencies (SLD) (Yairi and Ambrose, 2005).

3. Repetition and prolongations nature: There are three aspects to calculate this variable to distinguish between a normally disfluent child and a child with stuttering:
  - a) Children with normal disfluency usually have one or two extra repetition units. The child is more likely to be a stutterer when the number of repetition units is increased.
  - b) Rhythm of repetition: Children with slow and regular repetition are normal while children with rapid and irregular repetition are more likely to be stutterers.
  - c) Monitor symptoms of tension in repetitions and prolongations: Speech language therapist can observe the tension from the facial expression and hear the quality of the child's voice. Normal children are usually not stressed with their disfluency.
4. Airflow and phonation sustaining: Children that stutter usually have a problem on this point. Many children who stutter suffer frequent and abrupt starts and stop on certain words, particularly repeated words.
5. Physical indications of stuttering such as eye blinks, hand movements and head nods.
6. Word avoidance: This sign of stuttering happens when the child starts to say certain words and changes it suddenly, as though avoiding saying it.

If the child exhibits at least one of the previous characteristics listed above, they are considered a borderline stutterer.

### **2.3.3 Stuttering severity**

The purpose of this phase is to determine whether the child is normally disfluent and to decide the level of stuttering for children who do stutter. Instrument (SSI-3). The

SSI-3 is a tool used to measure the stuttering severity. There are four main levels that include: typical disfluency, borderline stuttering, beginning stuttering and persistent stuttering.

- **Typical disfluency:** To consider the preschool child as a child with normal disfluency, they must meet all the following criteria: 1-Less than 10 disfluencies per 100 words. 2-No more than two repeated units per repetition and the rhythm of the repeated units must be slow and regular. 3-The rate of the stuttering-like disfluencies (SLD) to total disfluencies must be <50%. 4-The child must be dealing with these disfluencies normally (if they aware of them at all).
- **Borderline Stuttering:** For the child to be considered as suffering borderline stuttering, they must meet at least one of the following criteria: 1-More than 10 disfluencies per 100 words, but the child is still relaxed and less tense. 2-More than two repeated units per repetition. 3-The rate of the SLD to total disfluencies is between 50% and 70%.
- **Beginning Stuttering,**The child is usually beginning to stutter between 3.5 and 6 years old. The key characteristic accompanying the stuttering however is hurry and tension. For the child to be considered as one that is beginning to stutter, they must meet some of the following criteria: 1-Stuttering is rapid and the child is visibly uncomfortable while speaking. 2-The child suddenly repeats words or phrases. 3-Pitch increases during prolongations and repetitions. 4-Aware of the stuttering and tries to avoid it. 5-Problems in starting air flow. 6-Facial expression tense. 7-The rate of the SLD to total disfluencies is >70%.
- **Persistent Stuttering,** Risk factors are the elements within the child or within the environment of the child which can increase or decrease the persistence of the stutter (Guitar & Guitar, 2003), while also influencing the level of treatment.

### **2.3.4 Disagreement on various assessments of stuttering**

One of the biggest disagreements during the assessment phase is whether the therapist should consider whole-word repetitions as primary symptoms of stuttering. Yairi and Ambrose (2013) include whole-word repetitions as a sign of stuttering while Cook et al. (2013) use a tool which excludes them. Howell and Davis (2011) conducted a study to create a model which predicts whether stuttering in children will continue to be persisting or recovering by teenage age. Severity measures were performed with inclusion/exclusion of whole-word repetitions on the predicted groups. The result indicates that all models that exclude whole-word repetitions were better than models that included this event.

However, the impact of excluding the whole-word repetitions and not using them during the diagnosis phase might be missing some children who do actually stutter and need help. In addition, it is likely to lead to a low recovery rate because it would possibly exclude some children who stutter (Howell, 2013). In contrast, the effect of including the whole-word repetitions and using them during the diagnosis phase may lead to treating children who are not in fact stutterers. This would also increase the recovery rate, as therapists would in effect be treating children who do not stutter.

Furthermore, there is another disagreement between speech language therapists about the determination of which risk assessments list that should be considered. The SLP may have difficulty in selecting the appropriate risk factors when diagnosing a person as a stutterer (Howell, 2013). Also, the therapist cannot depend on a single factor to determine whether a child will keep stuttering or not. Applying a set of factors could aid the therapist to decide which treatment is more suitable for this particular case (the child who stutters).

## 2.4 Treatment of stuttering

Determining the severity level of stuttering will help the therapists to decide which level of treatment is needed. The therapist will notify the parents that there is no need for treatment if the child has only normal disfluency. However, a comprehensive course with different sessions will begin if the child has a more serious indication of stuttering (borderline or beginning stutter). The following section describes two popular treatment approaches.

### 2.4.1 Treatment approaches

- **Direct and indirect approaches** The main contribution to these treatment models is provided by Van Riper and Johnson (Van Riper, 1973). Each model has targeted the specific concerns in the relationship between the child who stutters and their parents. The indirect approach relies on the assumption that the child's fluency issues might be solved by managing the environment rather than directly working with the child's speech. If the indirect treatment does not decrease the stuttering during six weeks, then this approach is considered to have failed. The causes of the failure however remain unknown. Sometimes, a family is unable to change the environment as planned or the child's stuttering increased for whatever reason. Thus, the direct approach is usually used for borderline stutters who have not responded to the indirect approach. The direct method defines as the bases of the recommendations to the parents of stuttering children on how they can deal with the first symptoms of their young child's stuttering without the child necessarily noticing, with the assumption that the child will actively follow the 'fine-tuning' speech attempts such as the lexical, syntactic and rate characteristics of the parental models. Also, the direct approach is used by clinicians

who do not decide to use the Lidcombe Program (Guitar, 2014), which is described next paragraph.

- **Lidcombe Program** This treatment approach is mainly designed for children who stutter when younger than six years old. This approach requires a high understanding, willingness to learn, and commitment from the parents. So, this technique of treatment is a collaborative program between the clinicians and the parents who work closely together to create daily chances for constructive responses to the child's fluent and stuttered speech at home, the child's natural environment. Treatment starts with formal discussions and quickly moves to informal discussions during the day. Once the child is fluent in all situations, the therapist manages to withdraw all clinical contact while carefully monitoring the progress. Then the family takes over carrying out and refining the treatment as needed. As mentioned, this program requires a large commitment from both parents and children. It also needs also a lot of warm and supportive feedback from the parents' side (Onslow and Packman, 1999; Guitar, 2014; Ratner and Tetnowski, 2014). **Lidcombe parent training elements** Usually, parents need specific training to apply the elements of the program. The following points will mention the main points that parents learn during the initial stages (Ratner and Tetnowski, 2014).
  - Distinguish normal fluency from unambiguous stuttering.
  - Rate the child fluency daily (from 1 to 10 scale).
  - Praise fluency.
  - Request corrections of stutters in a supportive manner.



### 2.4.2 Different opinions on stuttering treatments

The field of stuttering treatment has a range of different and diverse opinions. Behind each opinion is the voice of someone who believes that they are helping by providing their therapeutic methods and associated approaches. Even people who are not formally in this field may also believe that their suggestions are helpful for the people who stutter. They may believe this because the speaker is more likely to suffer less stuttering when they are trying different things. This is explained by the fact that any techniques that tend to highlight the events of stuttering will result in a reduction of the behavior (Ratner and Tetnowski, 2014).

Ratner and Tetnowski (2014) believe that effective treatment is not limited to decreasing the frequency of stuttering by those who suffer from the disorder. They also believe that successful treatment includes improving the quality of the stuttering, the quality of fluency and the quality of the patient's communication ability. Those that hold this point of view also believe that successful therapy should include steps on how to live an unrestricted life even while suffering from stuttering.

The correct understanding of stuttering is missed by most non-stutterers; while this is both normal and understandable, the real problem is that there exist professionals in the field who fail to completely understand this disorder. These professionals are unlikely to provide help that fully addresses the scope of the issue. Moreover, many clinicians see the stuttering in a categorical manner, in the way that a fluent person is assumed to be a good speaker while a stuttering speaker is considered to be inferior, rather than just different. Once these beliefs are acquired in the clinic, they tend to be reinforced in the child's home environment.

In general, fluency is better and stuttering does get in the way of communication but all talk of treatment aside, they are similar in every other way; it is possible to be a good communicator even while suffering from stuttering (Ratner and Tetnowski, 2014).

Ratner and Tetnowski (2014) disagree with Ingham and Cordes (1998) due to the assertion that fluency is the main issue in stuttering, although, they do agree with the idea mentioned

by Ingham and Cordes (1998) which states that the goal of stuttering treatment fundamentally is addressing and overcoming the self-judgment of stuttering and related behaviors; self-judgment and the opinion of the speaker about themselves is crucial.

In sum, while they disagree about certain things, a number of different experts agree that the most important thing in stuttering treatment is paying attention to what the clients need and the ways in which the clinician can provide help that will assist them to make the changes that they need.

## **2.5 Summary**

Stuttering is a complex speech disorder, and therapists have diverse opinions regarding its assessment and treatment (Ratner and Tetnowski, 2014; Yairi and Ambrose, 2005). One area of disagreement is whether therapists should consider whole-word repetitions as primary symptoms of stuttering. This argument is critical as it affects apparent recovery outcomes (Cook and Howell, 2013; Howell, 2011; Yairi and Ambrose, 2013). Each dissenting opinion is supported by the belief that the proposed therapeutic methods will help patients.

Ratner and Tetnowski (2014) believe that effective treatment should not be limited to decreasing the frequency of stuttering. They argue that successful treatment includes improving a patient's communication abilities. Ingham and Cordes (1998) state that fluency should be the main issue addressed by stuttering treatments. Nonetheless, most experts agree that the main goal of treatment should be to address self-judgment and related behaviors.

To summarise, stuttering treatment should be sensitive to clients' needs. Treatment involves a high degree of interaction between the stuttering child, his or her parents and the therapist. Therefore, stuttering is considered a complex speech disorder (Guitar, 2014; Onslow and Packman, 1999; Yairi and Ambrose, 2005).

From a computer scientist's point of view, this thesis has focused on which variables could be used to automatically analyze the speech sample such as frequency of disfluency

and dysfluencies types. Other variables in determining the pattern of disfluencies such as airflow and phonation sustaining and word avoidance could be only analyzed by the therapist. However, our current framework could help the therapists by providing a tool that automatically recognises stuttering events from the speech sample and provides a usefully written transcription of what was said. Also, it gives an indication of the severity level from the number of events in the speech sample. This service facilitates the analysis process of the speech sample.



# **Chapter 3**

## **Automatic Stuttering Recognition**

### **Background**

As seen in the previous chapter, clinicians need to measure stuttering events carefully during the assessment phase to determine the severity of stuttering. This task is critical and highly dependent on the clinician's experience. Unfortunately, no effective automatic tools exist to assist clinicians during the process of counting disfluency events. The focus of this thesis is to define and develop a system for recognising stuttering in children's speech. Such a system should automatically detect stuttering events from speech signals and produce full, verbatim utterances using modified automatic speech recognition (ASR). This chapter reviews work relating to recognition systems designed to detect or classify stuttering speech disorders; previous research has presented various methods and algorithms that have been applied to recognising stuttering events from speech signals.

#### **3.1 Introduction**

Stuttering is a disorder that can greatly affect communication due to frequent, disruptive disfluencies. Clinicians usually conduct disfluency counts in real-time while a child is talking,

to determine whether a child stutters, evaluate the progress of current treatment, or record treatment outcomes (Yaruss, 1997; Yaruss et al., 1998; Conture, 2001). This task is both critical and highly dependent on the clinician's experience (Blood et al., 2010; Brundage et al., 2006; Lass et al., 1989; Brisk et al., 1997), but as no record is taken of what was spoken, the speech cannot be re-examined to improve the quality of the assessment or evaluate its validity.

In a different approach, the clinician transcribes a recorded session and classifies each spoken term into one of several categories: normal, disfluent, or stuttering (Gregory et al., 2003; Guitar, 2014; Yairi and Ambrose, 2005). This approach provides more comprehensive and precise counts (Ratner et al., 1996), but takes both time (because of the need to transcribe every spoken word) and effort (because of the knowledge required of the relevant categories).

Unfortunately, no effective automatic tools exist to assist clinicians during the disfluency counting process. Research in this area has investigated two main approaches to detecting stuttering events. The first attempts to detect stuttering events from recorded speech signals in order to count disfluencies; the second involves transcribing the speech of children who stutter using ASR. Attempts at both are described in the literature.

ASR systems have been proven to be a useful technology which allows a recorded speech to be recognised and transcribed. However, it is well known that children's speech poses problems and is much more challenging for ASR than that of adults – previous research has reported poor performance of ASR systems when recognising children's speech (Liao et al., 2015a; Russell and D'Arcy, 2007). Moreover, conventional ASR systems are not designed or optimised for the task of recognising stuttering disfluencies, such as sound repetitions, and cannot be applied to annotate stuttering speech.

Because this thesis aims to provide an automatic means of classifying stutter severity, it will also aid in the detection and recognition of stuttering events. For this purpose, this thesis describes a system for the transcription of stuttering speech with a corresponding count of its

stuttering events. A count of stuttering events indicates the initial severity level. Ideally, the verbatim transcription will help clinicians to refine the assessment process and to store their data in a manner useful for future investigations.

The following two sections outline the development and implementation of established techniques for counting stuttering that use direct signal classification and ASR-based approaches. The direct signal classification approach uses machine-learning methods to classify stuttering-event segments directly from recording signals. In other words, no attempts are made to recognise precisely what words were being spoken as long as the type of stuttering events are correctly captured. In contrast, the ASR-based approaches are extensions of speech recognisers, and they attempt to extend the speech recognition to recognise pseudo word and non-word segments that occur during stuttered speech and then use transcriptions to detect and classify stuttering events.

## **3.2 Direct classification approach**

This section summarises different machine-learning methods that have been applied to stuttering classification. These methods use a direct speech signal to catalogue the stuttering events. Neural networks, support vector machine (SVMs) and other schemes have been trained to classify previously identified segments, such as stuttering and non-stuttering. However, if these machine-learning approaches are trained to detect an event within some continuous stream of other events, then the classification problem becomes a detection problem, which is much harder to solve than the former.

### **3.2.1 Artificial Neural Networks (ANN)**

Artificial neural networks have been defined in a number of ways (Kohonen, 1987; Widrow, 1988; Haykin, 1999). All researchers agree that neural networks consist of processing cells

called neurons; these neurons are then trained using input and output collections presented to the network, after which the network should be able to recognise patterns presented to it. This idea has enabled the solving of complicated problems such as pattern recognition, optimisation, and prediction.

Several studies have applied neural networks to stuttering classification to classify repetition and prolongation events or for classifying fluency and disfluency in stuttered speech.

Howell and Sackin (1995) proposed the first attempt at stuttering recognition. Their study used ANNs and focused on identifying repetitions and prolongations. The network of the ANN algorithm worked by automatically connecting neurons to each other to map between inputs and outputs and train the model to classify stuttering events; the basic idea was that the input vector of ANNs would be the autocorrelation function and envelope. Their best accuracy was 80%.

Howell et al. (1997) went into more depth, recording 12 samples of stuttering children. Their study also used ANNs and focused on classifying each word as fluent or affected by prolongation or repetition of either sound or word part. The proposed system depended on the stuttering features obtained from the acoustic signal to train the disfluency model. A pre-processing step was applied to all speech signals by hand segmentation processes, before being used as an input to the ANN to identify the boundaries of each word. Word and phrase repetitions were then removed. Numerous parameters were involved in the ANN, including the duration of the whole or part word, first and second part fragmentation, spectral measures of the first and second parts, and energies of the part word.

The best-performing parameters included word fragmentation, spectral measures, part-word duration, and energy. The results showed that 95% of the fluent words had been correctly identified. However, the classifier could only identify 43% of the repetitions and 58% of the prolongations individually. The difficulty in differentiating between the stuttering types may have been caused by variations in prolongation. Another point which might have



increased the complexity of the task was the fact that the ANN had no prior knowledge of the possible input. The network could therefore not apply foreknowledge of the typical acoustic features for that word to classify atypical characteristics and predict the correct stuttering type.

Czyzewski et al. (2003) proposed a classification model based on classifying stop-gaps, prolongations, and syllable repetitions. The study employed six fluent samples and artifacts of six disfluent samples. ANN classifiers and rough sets were applied to recognise stuttering events. The averaged recall achieved of ANN was 73.25%, while the averaged recall of rough set was over 90%. The results obtained from the rough set-based system was much better than the results achieved from the ANNs.

Szczurowska et al. (2006) examined the ability of neural networks classification between fluent and disfluent speech samples. Eight stuttering speakers were employed in the study, with the Kohonen technique and Multilayer Perceptron (MLP) networks applied to identify fluent and disfluent speech. The best result obtained was 76.67%.

Świetlicka et al. (2009) conducted research on automatic classification of a different type of disfluency in stuttered speech. Eight stutterers and four fluent speakers were involved in this study, and 59 samples of both fluent and disfluent speech, reading the same sentences, were collected from recording sessions. Radial Basis Function and MLP networks were used to distinguish and classify between fluency and disfluency in the samples. The results ranged between 88.1% and 94.9%.

Świetlicka et al. (2013) expanded this research to automatic classification of three types of stuttering disfluency (blocks, syllable repetitions, and prolongations). Their main idea was to build a system with hierarchical neural networks and evaluate it with respect to its ability to classify the three different stuttering events. The authors employed 153 pre-segmented four-second utterances containing these stuttering events from both fluent and disfluent speech, selected from 19 stuttering people and 14 fluent speakers.

For the first networks, they applied specific parameters to the speech samples and used them in an input set. The first network used was a Kohonen network, also known as a self-organising map (SOM). SOM was the first stage of the preparatory transformation of the input data, whereas MLP was the second stage of the classification of the examined utterances. The results achieved for the MLP networks were 68.6% for sensitivity and 97% for specificity, indicating that the classifiers could classify the three types of stuttering disfluencies on the tested data.

One concern with this method is that it cuts the speech recordings into 4-second segments, meaning that two important pieces of information are lost: where the prolongation happened and the number of prolongation events that occurred. According to Zebrowski (1994), the average prolongation event length is 706 msec. Therefore, a segment length of 4 seconds is much longer than the average prolongation event length, meaning that more than one prolongation event could occur in a 4-second segment. An accurate number of prolongation segments in the test sample cannot be determined by cutting the speech recordings into 4-second segments. Moreover, accurate durations for these prolongation events are also lost.

### **3.2.2 Support Vector Machine (SVM)**

The SVM is a powerful discriminative model with a collection of supervised learning methods for different objectives, such as classification, regression and detection. This classifier determines a linear hyperplane in the feature space that maximises the Euclidean distance ('margin') between the closest training samples of each class and the hyperplane. This step attempts to cover the widest possible error margin, and, consequently, the results of this classifier are very good for practical applications (Pedregosa et al., 2011; Cevikalp, 2017).

SVM has been used as a classification technique in stuttering classification. Ravikumar et al. (2009) introduced an automatic classification approach that aimed to classify syllable

repetition in reading speech. This research employed four phases: segmentation, feature extraction, score matching, and decision logic. The study applied an SVM as a classifier to the decision logic phase as an enhancement of the existing work by the same researchers, employing it to differentiate between fluent and non-fluent speech. The training set consisted of 12 samples collected from adults who stutter, while the test set had only three samples, which represented 20% of the dataset. The system achieved a high accuracy of 94.35%, achievable because the task was classifying between stuttered and non-stuttered speech segments.

### **3.2.3 Linear Discriminant Analysis (LDA) and k-Nearest-Neighbour (k-NN)**

LDA is a common technique that has been applied to data classification and dimensionality reduction (Balakrishnama and Ganapathiraju, 1998). Another fundamental and simple classification method is k-Nearest-Neighbour (k-NN) classification. K-NN classifiers are suitable for cases where there is no previous knowledge about the data distribution (Peterson, 2009). LDA and k-NN are widely employed for classification problems in speech.

Two studies have applied these classification methods to classify different speech disfluencies. Chee et al. (2009) implemented an MFCC feature extraction algorithm to classify prolongation and repetitions in stuttered speech, and applied LDA and k-NN as classifiers to evaluate the results, but they only used 10 samples from the UCLASS database (Howell et al., 2009). The prolongations and repetitions were manually segmented and used as an input to the feature extraction; the results showed a best accuracy of 90%.

Ai et al. (2012) presented a study which aimed to compare two feature extraction techniques, MFCCs and Linear Prediction Cepstral Coefficients (LPCC), to classify stuttered events. The researchers identified the stuttered events by manual segmentation and applied them to the feature extraction phase; 39 speech samples were chosen from the UCLASS.

The experimental investigation showed that the MFCC and LPCC features were able to classify stuttering events, and LDA and k-NN were applied to evaluate the LPCC and MFCC features in the identification of prolongation and repetition in stuttered speech. The same database was used for comparison between MFCC and LPCC; the experiments found that the LPCC features were slightly better than the MFCC features in different value parameters such as window overlapping and frame length selection. The highest results of LPCC with best configuration features has presented the accuracy of 94.51%. The best configuration of MFCC features has shown the accuracy of 92.55%.

### **3.2.4 Vector Quantization (VQ) framework, End-Point Detection (EPD) and Dynamic Time Warping (DTW)**

Other related studies have used the same feature extraction methods and classification techniques, but implemented their algorithms using different approaches such as VQ frameworks, EPD, and dynamic time warping.

A study by Mahesha and Vinod (2012) aimed to classify different kinds of stuttering disfluencies, such as repetition, prolongation, and interjection, relying on MFCCs and VQ frameworks. They generated a VQ codebook by gathering the vectors of training features of all types of stuttering disfluency using a k-means algorithm and saved them in a database for disfluency. The algorithm then applied a measure of distortion that reduced the Euclidean distance to link together the unknown stuttering disfluency and the identified stuttering disfluency in the database. This paper used UCLASS corpus, however, the number of applied speech samples is not mentioned. Instead, they mentioned the number of applied segments for training process which were 150 segments in total and 30 segments for testing. Finally, an analysis to assess quality was applied. The proposed algorithm produced a best accuracy of 86.67% for repetition, 96.67% for prolongation, and 100% for interjection.

Yeh et al. (2015) conducted a study on identifying three different types of repetitions: part-word, whole-word, and multi-syllable word. This approach was tested using artificial stuttering speech samples of Mandarin Chinese. Ten speakers imitated stuttering by recording 39 predefined repetition sets. The algorithm started by parameterising the speech samples and selecting six acoustic features which included volume, high-order derivatives, spectral entropy, and zero crossing rate. An EPD technique was then employed for speech segmentation and DTW was applied during processing of the segmented features to recognise similar patterns in neighbouring segments. The results showed that EPD can identify repetition in artificial stuttered samples; the results showed accuracy of 83%.

### **3.3 ASR based approach**

This section explores how ASR can assist in both diagnosing stuttering disorder and creating archives for further investigative research into this disorder. In the ASR-based approach, the input signal is not used to directly classify or count stuttering events. Rather, speech, including part-words and non-words utterances, is recognised and then examined.

ASR is already widely used in speech pathology as an assistive technology (Ward et al., 2016; Duenser et al., 2016; Middag et al., 2010; Fredouille et al., 2005). However, it is well known that children's speech poses problems for ASR; previous research has reported lower performance of ASR systems in recognising children's speech than adult's speech (Liao et al., 2015a; Russell and D'Arcy, 2007) caused by factors such as variable speech rate and small vocal tract length (Liao et al., 2015a). Despite substantial reported efforts to improve ASR for children's speech, progress in this area is still limited comparing to recognising adult's speech (Chandrakala and Rajeswari, 2017).

Detecting stuttering events in children's speech is even harder, and therefore all work in the literature has featured ASR systems built for reading speech, which is not as rich in disfluencies as spontaneous speech (Banerjee, Beck and Mostow, 2003; Banerjee, Mostow,

Beck and Tam, 2003). It is easier for ASR to recognise reading speech due to prior knowledge of what the subject intends to say, which limits the search space for predicting the next word. Furthermore, the reading speech task is already applied as a part of most stuttering assessments (Gregory et al., 2003; Riley, 1994). The following section provides an overview of the ASR system. Then, the following two sections demonstrate the application of ASR in different approaches to detecting and classifying stuttering events.

### 3.3.1 ASR overview

The idea of an ASR system is to convert an input speech signal to the most probable word sequence. Figure 3.1 presents the main parts of a general ASR system and shows the process of recognising the likelihood of most probable word hypotheses for the given speech input. The first step extracts the acoustic features by sampling the audio signal, and these feature vectors are then used as input in a decoding phase. A brief introduction to each element of the system follows below.

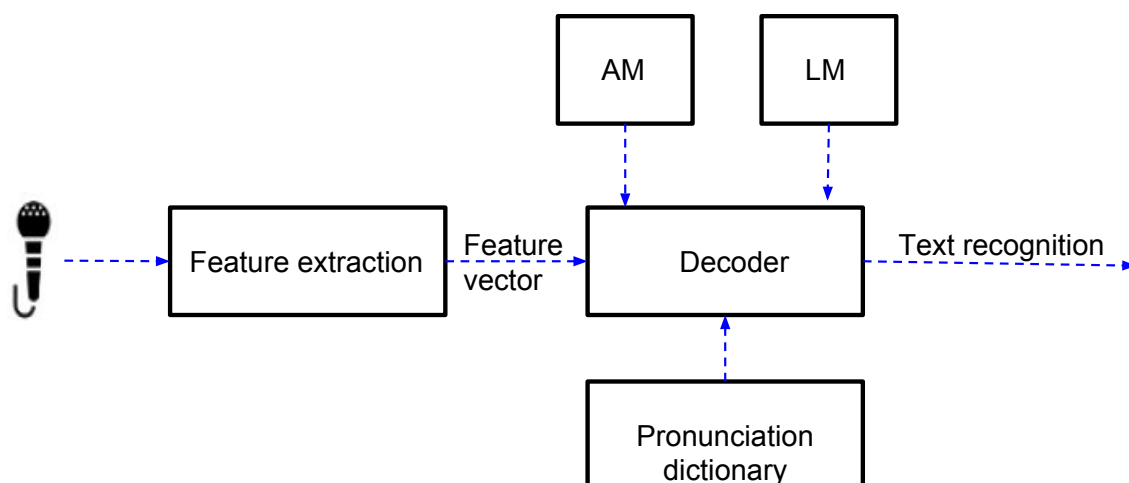


Figure 3.1 *Standard ASR system architecture.*

### Feature extraction

The first stage in the speech recognition system is known as feature extraction, or front-end processing, which begins by pre-processing input speech signals to provide a stream of observations  $O = o_1, o_2, \dots, o_t$  of *fixed size acoustic feature vectors*. The purpose of the feature extraction stage is to define compressed observations to aid the recognition task. Two speech feature extraction techniques are widely applied in state-of-the-art ASR systems: *Mel-Frequency Cepstral Coefficients* (MFCC) (Davis and Mermelstein, 1980) and *Perceptual Linear Prediction* (PLP) coefficients (Hermansky, 1990). There is a particular process called segmentation which can be applied in some domains when the front end is also required to separate speech segments from the entire audio stream. This process can be used in different application such as broadcast news transcription and telephone speech recognition.

### Decoder

The second stage in the ASR system is decoding the features. The main aim of the decoder or inference component is to analyse the extracted sequence of acoustic feature vectors to recognise the sequence of words  $W = w_1, w_2, \dots, w_n$  given the acoustic observations  $O$ . In particular, the decoder attempts to determine the following:

$$\hat{W} = \underset{w \in \mathcal{V}}{\operatorname{argmax}} P(W|O) \quad (3.1)$$

Then, Bayes' rule is applied to convert the above equation to the equivalent problem

$$\hat{W} = \underset{w \in \mathcal{V}}{\operatorname{argmax}} \frac{P(O|W)P(W)}{P(O)} \quad (3.2)$$

$P(O)$  is fixed acoustic data and does not change over the recognition process. Therefore, the search problem can be divided to two main parts: acoustic modelling  $P(O|W)$  and language modelling  $P(W)$  (Rabiner et al., 1993):

$$\hat{W} = \underset{w \in V}{\operatorname{argmax}} \underbrace{P(O|W)}_{\text{Acoustic model}} \underbrace{P(W)}_{\text{Language model}} \quad (3.3)$$

The ASR system tries to find the most likely word sequence  $\hat{W}$  (hypothesis transcription) in a time-synchronous manner using a combination of different knowledge resources (e.g. acoustic and language models, lexicon) and given a sequence of acoustic observations.

Decoding can be efficiently undertaken using two commonly applied algorithms: stack or  $A^*$  decoding (Martin and Jurafsky, 2009) and time-synchronous Viterbi decoding (Forney, 1973). These mainly attempt to find the best path with the highest likelihood by expanding words, phonemes and hidden Markov model states, and collecting scores from the different components. Once the algorithm reaches the end of the given acoustic observation sequence, a traceback process of the path with the highest likelihood is layered to produce the best hypothesis transcription (1-best output). Usually, implementation uses pruning methods (Lowerre, 1976) because of the vast search space to discard unlikely branches and reduce computational cost.

The rest of this section briefly describes each of the source knowledge components used in the decoder along with the evaluation metrics for system performance.

### **Pronunciation dictionary**

The pronunciation dictionary, also known as the lexicon, is a word list with relevant pronunciations given as a sequence of phones; a phone is the basic unit of sound. Therefore, the pronunciation dictionary is applied to map phones to the words that are employed in the language model. Moreover, each word in the pronunciation dictionary can have more than a



single pronunciation. There are different publicly available pronunciation dictionaries that can be applied for building ASR systems, including the American English Carnegie Mellon University Pronouncing Dictionary (CMU, 1998) with about 134,000 words and the British English Example Pronunciation Dictionary (Robinson, 1996) with about 250,000 words.

### **Acoustic model (AM)**

The AM describes the relationship between the acoustic knowledge represented by the extracted feature vectors and the word sequence. This acoustic knowledge is used to find the likelihood of  $P(O|W)$  in Eq. 3.3, where  $O$  is an observation of the feature vectors sequence and  $W$  is the words sequence. This likelihood is obtained from simultaneously feeding the model with these acoustic feature vectors and with the original textual transcriptions. To achieve this, each time frame in  $O$  is identified and a likelihoods vector for the frame is produced by each possible word in  $W$ . To enhance the estimation reliability for each word in  $W$ , the number of transcribed training samples can be increased. However, obtaining transcribed speech data is expensive. Consequently, ASR systems apply sub-word units as the acoustic models since they are more likely to be recognised than a complete word. Usually, the sub-word units are context-dependent phones. The unit applied in the AM will here be referred to as an acoustic unit. During the process of using sub-word units, a pronunciation dictionary is needed to map a word to similar sequences of sub-words.

Most state-of-the-art systems, including large vocabulary continuous speech recognition (LVCSR) systems use a hidden Markov model (HMM) in their AM core to capture the variability in the speech represented in the acoustic signal (Rabiner et al., 1993). HMMs are applied in LVCSR systems to model sub-word units (i.e. monophone or triphone models). Subsequently, the sub-word units are accumulated to produce word-HMMs based on the rules defined by the pronunciation dictionary. The HMM approach has been shown to

have powerful performance in ASR systems, and they can be implemented using different algorithms, such as Viterbi (Forney, 1973) or Baum-Welch (Baum et al., 1970).

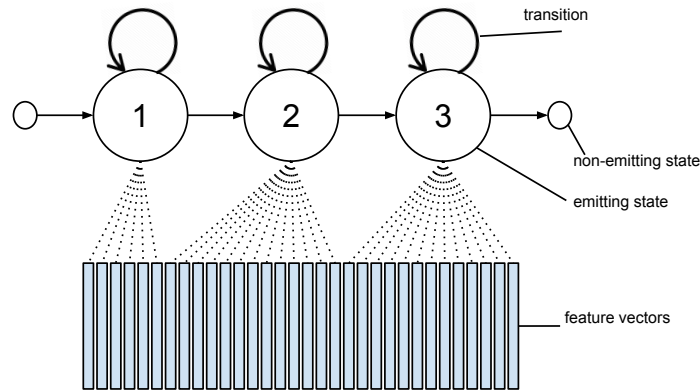


Figure 3.2 An HMM model that includes three emitting states and both entry and exit of non-emitting states.

Figure 3.2 presents a left-to-right HMM example. Each state is connected with weighted mixtures of Gaussian distributions. An HMM model including emitting states ( $N$ ) constitutes the following components:

- Transition probabilities,  $Z = [z_{ab}]$ ; where  $a, b \in \{1, 2, \dots, N\}$  that represent the transiting probability from state  $a$  to state  $b$ , so  $P(s_b | s_a) = z_{ab}$ .
- Observation probability distribution,  $P(x_t | s_i)$  which is usually represented using a distribution of multivariate Gaussian mixture for each state  $i$ :

Alongside Gaussian mixture model HMM systems (GMM-HMM), several methods that apply artificial neural networks (ANN) (e.g. (Seide et al., 2011; Yu and Seltzer, 2011)) have become common for producing a more reliable performance for different tasks (Hinton et al., 2012).

Different ANN approaches, such as multi-layer perceptron, deep neural networks and recurrent neural networks, are combined with HMM models in the system architecture. This

combination can be applied in tandem where the HMM-GMM model is trained on the top features produced from an intermediate, or bottleneck, layer within the ANN (Hermansky et al., 2000; Weninger et al., 2011; Hinton et al., 2012). The combination can be implemented in hybrid architectures where statistics are estimated from the HMM model with phoneme posterior probability produced in the output layer of the ANN. In this approach, HMM parameters are optimised with those of the ANN (Bourlard and Morgan, 2012; Parveen and Green, 2002; Dahl et al., 2012).

### **Language model (LM)**

The LM is a fundamental element of an ASR system. The role of the LM is to guide the search to decide the most likely word sequence by measuring the validity of a probable sequence in a provided language for a given domain (Rabiner et al., 1993).

The LM represents prior knowledge about the syntactic and semantic information of word sequences, and it can enhance the performance of ASR recognition by providing sufficient contextual information. However, the main challenge that faces language modelling is coverage of adequate syntactic, semantic and pragmatic aspects of a given language in a specific task domain. In Eq. 3.3, the prior probability  $P(W)$  of word sequence  $W$  describes the semantics and syntax of the given language.

The LM is independent of the acoustic model, and its parameters can therefore be established using extensive textual sources including journals, newspapers and web content. It is almost impossible to build LMs using pre-defined sets of linguistic rules because of natural complexity in the spoken word. As a result, the dominant approaches for constructing LMs have traditionally been statistical language modelling, such as n-gram model estimation, and, more recently, a recurrent neural network language model. Statistical language models are usually created from large volumes of training data obtained from the specific domain in which the model is to be used; detailed aspects of two methods to building LMs.

- **N-gram model estimation**

The most popular LM for speech recognition tasks is based on n-gram estimation (Bahl et al., 1983). This type of LM relies on the hypothesis that the probability of a word,  $w_i$ , depends on the preceding word,  $n-1$ . Therefore, for a given sequence of words  $(w_1, \dots, w_n)$  and applying the chain rule hypothesis, the estimation of the LM becomes

$$P(w_1, \dots, w_n) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1, \dots, w_{n-1}) \quad (3.4)$$

$$\approx P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_{n-2}, w_{n-1}) \quad (3.5)$$

$$\approx P(w_1)P(w_2|w_1) \prod_{i=3}^n P(w_i|w_{i-2}, w_{i-1}) \quad (3.6)$$

In Eq. 3.4, unigram and bigram refer to the first two terms while the last is a trigram since the two previous words are employed for the condition. Higher n-gram orders can also be applied, for example 4- or even 5-grams. The n-gram model parameters are estimated and performed using simple maximum likelihood calculations from the training data:

$$P(w_i|w_{i-1}, w_{i-2}) = \frac{\text{count}(w_{i-2}, w_{i-1}, w_i)}{\text{count}(w_{i-2}, w_{i-1})} \quad (3.7)$$

$$P(w_i|w_{i-1}) = \frac{\text{count}(w_{i-1}, w_i)}{\text{count}(w_{i-1})} \quad (3.8)$$

$$P(w_i) = \frac{\text{count}(w_i)}{\sum_{w_i \in V} \text{count}(w_i)} \quad (3.9)$$

where count  $(w_{i-2}, w_{i-1}, w_i)$  refers to the number of times that word sequences  $w_i$ ,  $w_{i-1}$  and  $w_{i-2}$  are observed in the training data. N-gram LMs are easy to build, although some of the word sequences may not be seen in the training text, making their probability zero. This creates a problem for the ASR which, consequently, will never recognise the word. Smoothing methods that involve discounting and back-off have been proposed to address this problem (Jelinek, 1997), for example, modified Kneser-Ney smoothing (Chen and Goodman, 1999) employs various discount parameters to provide effective results for low-order LMs.

- **Model interpolation** The most common approach for adapting a background model to a specific task domain is model interpolation which involves considering the weighted sum of probabilities provided by the model components.  $P(w|h)$  is the observed probability of a word,  $w$ , given the previous word sequence and its history,  $h$ . Consider a background model  $P_y(w|h)$ , and a specific domain-adapted model  $P_x(w|h)$  where  $\lambda$  assists as the interpolation co-efficient and  $0 \leq \lambda \leq 1$ . The desired model  $P(w|h)$  can be achieved thus:

$$P(w|h) = (1 - \lambda)P_y(w|h) + \lambda P_x(w|h) \quad (3.10)$$

A popular context for applying LM adaptation is when the availability of target-domain data is limited while other domains include large volumes of data. In this situation, the target-domain model is merged with the background model (other-domains) via linear interpolation. Commonly, a tuning process is applied to the interpolation co-efficient by minimising the perplexity on similar data to the target domain (test/development dataset) (Jelinek et al., 1977). Here, Eq. 3.10 is generalised and used to combine different pre-defined domain-specific LMs (Bellegarda, 2004). The mixture model

probability of specific-domain models, ( $P_Z$ ) is provided as:

$$P(w|h) = \sum_{Z=1}^Z \lambda_Z P_Z(w|h) \quad (3.11)$$

where  $Z_{th}$  is the specific domain model and  $\lambda_Z$  indicates its interpolation co-efficient. Several sources of knowledge can be employed to determine the best interpolation co-efficients for  $Z_{th}$  LMs in a given task. These strategies have frequently been used to dynamically adapt a background model using some knowledge about variations in the specific speech domain (Echeverry-Correa et al., 2015). Figure 3.3 presents an example set-up for devising dynamic language models through comprehensive model interpolation.

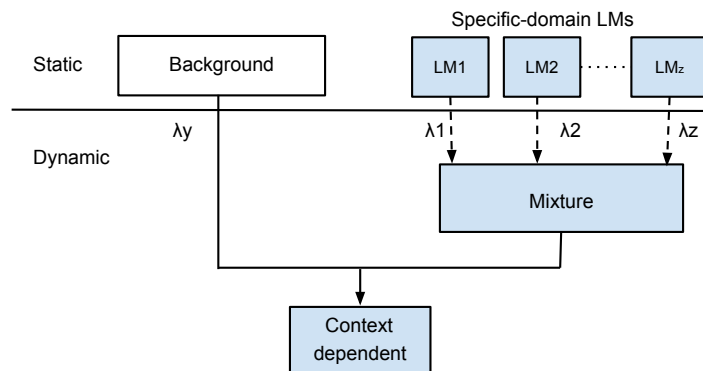


Figure 3.3 *Construction process of a context-dependent LM through interpolation of an out-of-domain model with in-domain changes in speech.*

### ASR Performance

Most speech recognisers of large vocabularies that apply fast decoding algorithms provide near real-time responses, achieved by high-performance systems that offer increased computational power. In addition, significant recent achievements in employing deep learning

in speech recognition mean that current advanced ASR systems have been trained to attain human-level recognition performance in conversational speech (Xiong et al., 2016).

However, no modern ASR system is 100% correct; a small gap always exists in real-life scenarios between the accuracy of humans in recognising speech and the accuracy of current ASR systems. This gap becomes clearer when the system is used to recognise speech in noise or spontaneous speech.

To evaluate the performance of ASR systems, the best hypothesis transcription is compared with a reference transcription, and the word error rate (WER) is usually used as the standard evaluation metric in this context. The WER is a primary measure of how significantly the hypothesised transcription returned by the ASR differs from the reference transcription, and is calculated as the minimum number of edit operations—substitutions (S), word insertions (I) and word deletions (D)—that are necessary to map the hypothesised and reference transcriptions. Note that the error rate can be greater than 100% due to insertions. The WER is thus defined as:

$$WER = \frac{Deletions + Substitutions + Insertions}{TotalWords} \times 100 \quad (3.12)$$

<b>REFERENCE:</b>	this	path	is	the	***	easiest	one
<b>HYPOTHESIS:</b>	this	***	is	the	only	easiest	way
<b>Evaluation:</b>	C	D	C	C	I	C	S

Figure 3.4 An example word error rate evaluation.

An example is shown in Figure 3.4 which presents a reference sentence, the hypothesised version and values for each word error type. The WER calculation of the given example is

$$WER = \frac{1 + 1 + 1}{6} \times 100 = 50\%$$

There are other metrics called *phoneme error rate* (PER) and *grapheme error rate* (GER) which relied on the unit applied transcribe the reference. These metrics can be calculated by comparing the acoustic units of hypothesised transcriptions against matching units in the reference transcription using an equation equivalent to Equation 3.12.

The *Matched-pairs* test is a standard statistical test for comparing the performance of ASR systems based on WER (Gillick and Cox, 1989), which examines the number of word errors between the two versions, averaged over a number of segments. The official implementation of WER and *Matched Pairs Sentence-Segment Word Error* (MAPSSWE) test is provided as the free script 'sclite' by the *National Institute of Standards and Technology* (NIST) (Kramida et al., 2018). However, most recent speech recognition toolkits, such as Kaldi ASR (Povey et al., 2011), already include the test.

### 3.3.2 Applying ASR for classification task

Two 2007 studies by Wiśniewski et al. (2007a,b) outlined an automatic stuttering classification system. These studies applied hidden markov model (HMM) models with different numbers of states to differently sized codebooks. HMMs are widely employed in speech recognition studies because speech signals can be treated as a short-time or piecewise stationary signals (Rabiner, 1989). Therefore, they can be used to identify some stuttering events. Different sample Polish-language sets were used for classifying prolongation and blocking events, and MFCC was applied as a feature extraction method. A recording sample length of 87.624 seconds was used for testing the application. The sample included 14 prolongations and 10 blocking stuttering events. The best results, achieved after removing silence, gave a sensitivity of 75% and a correctness of 70%. Essentially, this study preformed a classification of pre-segmented speech segments within continuous speech using a codebook system.

Tan et al. (2007) presented a research project that outlined the development of the Malay Speech Therapy Assistance Tools (MSTAT) system to help therapists identify different events



for children who stutter. MSTAT includes a speech recognition system that uses the HMM to evaluate speech difficulties in stutterers. The study asked the children who participated to read selected words loudly in order to identify a stutter. The dataset consisted of 20 fluent speech samples and 15 artificial stutter speech samples; 10 samples of both fluent and artificial stutter speech were employed for the training set, while the remaining samples were used for the test set. The results achieved for normal speakers and artificial stutter speech were 96% and 90%, respectively.

### 3.3.3 Applying ASR for detection task

One approach for automating disfluency counts using speech recognition was implemented by Nöth et al. (2000), whose system used a variable grammar that modelled the positional regularities of different stuttering events during a reading speech task, with a deterministic grammar for each phoneme. Phonemes were constructed as nodes with connecting edges that led to possible follower phonemes; edges were added to each phoneme where a possible stutter could occur. For example, they built filled-pause and silent nodes after each phoneme due to the possibility of these particular events (disfluencies) occurring after each phoneme.

However, the reported evaluation of the system omitted essential details – no information was given about the weights assigned to each edge, which could lead to increased false alarm or miss rates. Moreover, the work did not report the precise recognised stuttering events nor the false alarm rate. Without the miss and false alarm rates, the system's recognition accuracy is unclear, and without a count of correctly recognised stuttering events, the system cannot be used to measure stuttering severity, which is one goal of the work.

Hollingshead and Heeman (2004) built a system to automate stuttering counts similar in structure to the one demonstrated by Nöth et al. (2000). They built several grammar-based language models to capture stuttering events using uniform transition weights. Each grammar had three components, for modelling sound, word, and phrase repetitions, and each grammar

was applied to test the speech recogniser's performance in the CSLU toolkit (Sutton et al., 1998). The acoustic model is trained on fluent speech not on a stuttered speech. The selected test text contained 238 words, including 29 stuttering events. They divided the speech sample into 36 utterances by adding boundaries according a pre-determined locations.

Results indicated that these grammars recognized repetitions poorly, especially phoneme repetitions. The grammars were affected by high false positives rates, particularly for sound repetitions. One suggestion that is applying a statistical language model (LM) with specific transition probabilities could capture the repetitions and limit the false positives rate.

Studies undertaken by Heeman et al. (2011, 2016) merged ASR outputs with the clinician's own manual annotations to produce corrected transcripts of the stuttering speech. This study aimed to help clinicians spend less time correcting transcriptions and more time analysing the client's stuttering patterns. They observed that the quality of the word transcription can be improved by combining a clinician's annotations by a relative factor of 7.5%. Such an approach could not, however, be described as fully automatic.

### 3.4 Summary

This overview of the literature outlined some of the models that have been proposed for predicting stuttering events, such as repetitions and prolongations. Different classifiers can be employed, and the method of applying these classifiers differs between models. Many studies achieved a high accuracy rate as the task they attempted was classification rather than recognition – that is, differentiating between stuttered and non-stuttered speech segments (Ravikumar et al., 2009; Szczurowska et al., 2006; Świetlicka et al., 2009, 2013).

A common observation from most studies is the uncertain reliability of the achieved results because of the small sizes of the training and test sets and the lack of complete statistical findings. Reporting accuracy as the only reflective measurement in a study using a very small data set to train classifiers is not sufficient. Stuttering events are rare compared to

fluent speech, and training a classifier on unbalanced data would therefore lead to predictions of the dominant class, which in this case is fluent speech. The significance, therefore, cannot be determined, and studies mentioned in this survey such as (Chee et al., 2009; Ai et al., 2012; Mahesha and Vinod, 2012; Yeh et al., 2015) would need to report a breakdown of their results to reflect the actual performance of their systems.

In some respects, this survey is inconclusive and cannot definitively state that one approach is better than the other because some studies mentioned did not report their full results. More training data are needed to obtain a reasonable recall and precision rather than just a high accuracy, which does not reflect the correct classification of stuttering events. To ensure good practices, we will declare the amount of training data used, how the training and testing data were applied and provide statistical information.

The goal of this thesis is to build such an automatic tool that helps clinicians by providing them with a full verbatim transcription that includes stuttering events and a count of each stuttering disfluency to provide a more accurate evaluation of the system. The verbatim transcription could help clinicians to expedite the assessment of children's speech and make the diagnostic process more efficient. Also, it could be easier to archive data for further evaluation.



# Chapter 4

## Word Level Transcription and Annotation of Stuttering Data

A part of this work is to build a system based on either a direct machine-learning or an automatic speech recognition (ASR) approach, and, in either case, a sufficient volume of data is required to train this system. Therefore, this chapter discusses the problem of gathering data on children who stutter.

Publicly available datasets on children's speech are limited, even fewer exist for children who stutter. Namely, the only available corpus for children who stutter in English is the UCL Archive of Stuttered Speech (UCLASS) (Howell et al., 2009). However, UCLASS does not provide transcriptions of all available recordings. In addition, the transcribed data are not time-aligned, which limits their ability to be used for training an ASR system. In response, this thesis provides much-needed data in the form of verbatim and time-aligned transcripts. This chapter explains the transcription process for speech recordings from UCLASS, Release Two. Then, it describes the annotation process for each word, done with the purpose of training a classifier that automatically detects different stuttering events on ASR transcriptions using state-of-the-art machine-learning algorithms.

## 4.1 Introduction

Automatic speech recognition (ASR) systems face many difficulties and challenges when recognising children's speech, for a variety of reasons. There are particular problems in obtaining sufficient data to train ASR for children speech. Therefore, it is more difficult to train an ASR system for recognising children's speech than training an ASR system to recognise adult's speech. One of the main reasons is a lack of available data for children's speech. PF-STAR (Russell, 2006) is a commonly applied British English children's speech corpus that includes roughly 7.5 hours of reading speech. The size of the PF-STAR corpus is about 2.5% that of the Switchboard training set (John and Holliman, 1993), and one-tenth the size of the WSJCAM0 (Robinson et al., 1995).

Obtaining a public corpus for children with a stuttering disorder is even more challenging than one for children with typical speech, and the amount of data currently available is much smaller. The only available corpus for children with stuttering in English is Howell et al. (2009). Moreover, there are two main limitations of this corpus: there are no ready transcriptions available of the samples for training an ASR, and the recordings contain environmental and other background noises.

There is a need for a full verbatim/time-aligned transcript to train an ASR modelled to recognise stuttering events within children's speech. This chapter explains the process of making a full verbatim transcription for read speech samples in UCLASS, Release Two. There is also a demand for a word-level annotation to evaluate the ability of machine learning classifiers to detect stuttering events in transcriptions produced by the ASR.

This chapter is structured as follows: Section 4.2 introduces the UCLASS corpus and presents a detailed overview. Section 4.3 addresses the motivation for this task and its main contributions. The proposed transcription scheme is presented in Section 4.4. Section 4.5 describes the applied word-level annotation scheme. The intra-annotator agreement is

measured using a commonly known metric, the Cohen's kappa coefficient (Fleiss and Cohen, 1973) in Section 4.6. Section 4.7 provides a summary of the given work.

## 4.2 UCLASS: Children corpus overview

UCLASS stands for 'UCL Archive of Stuttered Speech'. This corpus is supported by the Welcome Trust and is split into two main releases. Release One contains 138 recordings of monologue speeches from speakers between the ages of 5 years, 4 months and 47 years. In total, 125 participants were children under the age of 18 years. The data set provides orthographic transcriptions (which are not time-aligned) for only 31 recordings. Figure 4.1 shows an example of an orthographic transcription that is provided in the corpus.

I've chosen to talk about my erm electric guitar, [A An] I've been playing it since [LARSSSSST] November, about that. And um, yeah well, I've been playing it for [JJJJJJ] JUST over a year now. An [um uh, HU] I did have a tutor, erm but in about, H AROUND September um he just {blocks}um SSS STOPPED coming, didn't turn up for some reason, and WO [W WU] we found out [A A] couple of months ago his wife's got cancer, so um so [HEE] HE'S going to stop for a bit. And um, well HI I HA [A ASN'T] been as fun um playing since he hasn't been teaching me, because when I learn songs by myself it takes a bit longer, and isn't quite as fun. And [um] well, [M MOST] of the SO SONGS I play is like rock, and hard rock, and um my favourite band are {blocks}HU U OBABLY [QUEEN QUEEN], and um, HI HIXEPT that I find most of their songs [QUITE QUITE] hard to play. Huh, my teacher said I should start off with SU ZOMETHING would be a bit more simple, AND [HA AND] we had just started moving [ON ON] to Queen, but he um stopped. And erm, Y EAH, and [AH] like playing U ALMOST any [um RRRROCK] bands, like Aerosmith and, and people like that. And um, I play in a band as well, except um, [A A] don't play the LEAD I HA [A ASN'T] LEED GUITAR in my band I play the bass, and we practice about [WW WWW] ONCE a week, round um one of our friend's house. And um, yeah OUR OUR [FFFFFRIEND] plays the drums and we have a lead guitarist, [A AND] a rhythm, which I was doing, and [HA HA] WE'VE only been playing for about a month now, and um, well there U U two other {block}GUITARISTS, who in the band also have the same tutor um [U UZ] me, and, erm.

Figure 4.1 *Orthographic transcription example provided by UCLASS (Howell et al., 2009). The red boxes have been added to highlight certain stuttering events by the author of this thesis.*

The Release Two recordings contain different types of speech: 82 monologue samples, where 80 samples are for children under the age of 18 years, 128 conversational samples, where 127 samples are for children under the age of 18 years, and 107 read speech samples, where 103 samples are for children under the age of 18 years. The data set provides orthographic transcriptions (which are not time-aligned) for only two read and four monologue

Table 4.1 *The UCLASS overview information about each release. The number of recordings of each category is provided in column number two. Age range and the mean age for each category are given in NNyNNm format, where y is the year and m is the month (Howell et al., 2009).*

Category	N	Age(Range)	N(under 18)	Age(Mean)	Male	Female
Release 1(monologue)	138	5y4m –47y0m	125	13y2.86m	120	18
Release 2 (monologue)	82	7y10m –20y1m	80	12y2.9m	76	6
Release 2 (read speech)	107	7y10m –20y7m	103	13y0.53m	92	15
Release 2 (conversation)	128	5y4m –20y7m	127	12y2.71m	110	18

recordings. There are also two monologue recordings that were transcribed phonetically.

Table 4.1 presents the main information about these two releases.

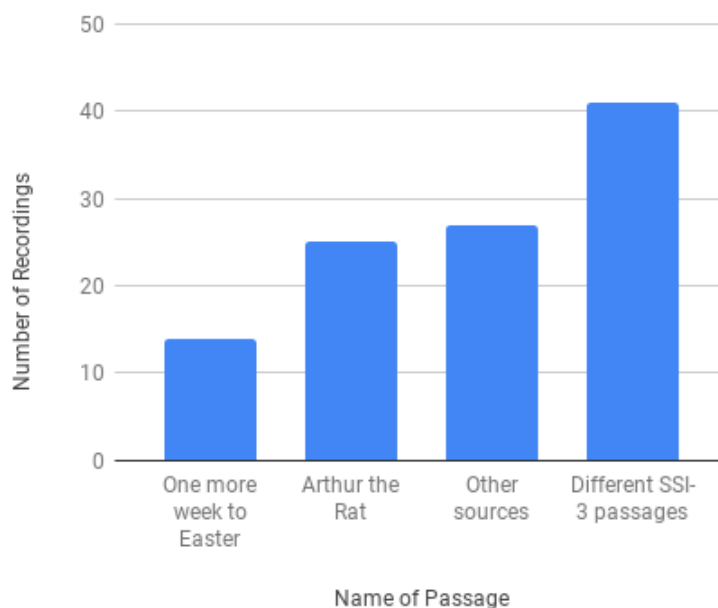


Figure 4.2 *Graph of different reading passages included in the dataset. Name of each reading passage is provided along with its number of recordings. The Stuttering Severity Instrument-3 (SSI-3) contains texts suitable for the age of the participants (Riley, 1994).*

The focus of this thesis is on read speech, as one of the targets of the project is to build an ASR for children who are asked to perform a reading task as part of a diagnostic session. Release Two provides 42 texts read by children from the stuttering severity instrument (SSI-3) text readings, which is suitable for the age of the participants (Riley, 1994). Another 25



recordings feature the reading of a passage from ‘Arthur the Rat’ (Abercrombie, 1964), 14 recordings include a passage from ‘One More Week to Easter’ which is a text developed in UCL lab (See Appendix B, for the given text), and 27 come from other sources. Figure 4.2 presents the name of the passages and the number of readings recorded for each of them.

The recordings vary in quality; some were made in noisy or low-quality sound environments. The main reason for the acoustic noise can be traced to microphone problems and/or tape hiss. Moreover, environmental noise affects the quality of the recordings, as some include noises from inside and outside of the room. Sounds from outside the room can include people moving in corridors or birdsong. Extraneous noises from inside the room include doors shutting, tables being bumped, items being dropped and the child moving and touching the cloth that covers the microphone. These types of noise present differently when recorded.

## **4.3 UCLASS: Contributions**

### **4.3.1 Overview**

The transcription process was undertaken to train an ASR system. Initially, the process began with a focus on spontaneous speech, particularly Release One monologues as non-aligned orthographic transcriptions of 31 out of the 138 recordings were already available. However, more data are required to train an ASR system. Earlier, it was assumed that it is possible to build and train an ASR system for stuttering recognition on spontaneous speech by applying an automatic alignment tool during training to fix the alignment issue. Therefore, out of the 138 recordings, the author transcribed another 32 recordings with a transcription approach similar to Howell’s. Only 32 recordings were transcribed due to difficulties in understanding what the spontaneous speech was, as the recordings were of children who stutter.

After an initial pilot study to build an ASR system that was trained on non-aligned orthographic transcriptions of spontaneous speech, two major issues were isolated. The first is related to the speaking style. Because the speaking style is spontaneous and training data are limited, most of the spoken words in the test set were not included in the training set. The second is related to the alignment problem. Using stuttered speech caused difficulties in the automatic alignment process between the orthographic transcriptions and the corresponding signals. In response, the task instead focused on read speech and the creation of time-aligned transcriptions using Audacity software to time-align different utterances of read speech from UCLASS, Release Two.

After applying the time-aligned data, a second pilot study was conducted to build and create an ASR system capable of recognising stuttering events. The baseline ASR of the pilot study recognised a very limited number of different stuttering events, such as word and phrase repetitions. However, no sound repetitions were recognised in the test set. Notably, Howell's transcription of sound repetitions, which used only initial constants such as in 'm m make', was not handled well by the orthographic to phonetic pronunciation dictionary rules. Therefore, Howell's orthographies and time-aligned transcriptions were altered, as a letter representing a schwa vowel for every single sound repetition was inserted accordingly. By convention, we chose the letter 'a' as our orthographic vowel, although the dictionary phonetically rendered it as a schwa. Because of this alteration, sound repetitions were properly recognised in the second pilot study, as a result of increasing their probability to non-zero.

There are two main contributions in this chapter. The first is a full verbatim/time-aligned transcription of a complete reading subset for UCLASS, Release Two. The other contribution is a word-level annotation that includes the stuttering type of each word.

The further study of ASR systems in this thesis is based on a standard reading task used by clinicians to diagnose stuttering in children. As previously noted, there is a need for a

full verbatim/time-aligned transcription of stuttered speech to train an ASR system. For this purpose, we obtained and transcribed all recordings of children’s read speech from UCLASS, Release Two (Howell et al., 2009). It contained 107 recordings of readings contributed by 40 different speakers. Section 4.4 provides a description of the transcription process.

To train machine learning classifiers to detect stuttering events from transcriptions produced by ASR, we annotated the data to include the stuttering type of each word following the annotation approach of Yairi and Ambrose (2005). We annotated all the reading transcriptions from Release Two without alignment information. In addition, we considered 31 out of the 138 non-aligned orthographic transcriptions from UCLASS, Release One that were made by Howell and the other 32 transcriptions that we made, as described previously. We annotated the data to include the stuttering type for each word. The annotation approach is provided in Section 4.5.

## 4.4 Transcription scheme

The UCLASS corpus does not provide a time-aligned transcription for each speech sample. Therefore, for the purposes of this thesis, a manual reference with word timings was created. The UCLASS corpus provides unaligned orthographic transcriptions for 31 samples in Release One. Howell’s method for transcribing stuttered speech was followed during the production of these transcripts. The same guidelines for transcribing stuttered speech were used for the rest of the read data (107 read samples). In addition, AMI guidelines (Moore et al., 2005) were applied to non-stuttered speech and full-verbatim transcription.

As mentioned previously, the schwa sound was added to the transcripts for all sound repetitions. This step was motivated by Howell and Vause (1986) findings that an occurring vowel often sounds like a schwa in the stuttered repetitions of a syllable, even when this schwa is not intended. The frequency of the schwa sound probably results from it being the most easily and readily articulated vowel sound. In syllable repetition, a schwa usually occurs

when an individual begins to vocalise while breathing. As in rapid repetitive blocks, there is insufficient time to make the articulatory movements necessary to produce the appropriate vowel of the repeated syllable. Hence, the more readily formed schwa tends to precede the appropriate vowel (Sheehan, 1974).

The full verbatim/time-aligned transcription process was applied using Audacity software to all read speech samples from UCLASS, Release Two. Audacity allows users to accurately select the start and end points for each utterance. Figure 4.3 presents an example of the Audacity-mediated transcription process. Once complete, the labels were exported to generate a file featuring full information for each utterance. Figure 4.4 shows an example of the final exported file.



Figure 4.3 This figure demonstrates part of one recording transcription process using Audacity software. Because of Audacity's simple interface, a user can accurately select the start and end points of each utterance as well as scroll right/left through the audio track.

80.965868	85.872891	I had been <b>va</b> very hard on Alice
85.872891	87.243971	is a <b>more</b>
87.604781	88.903699	<b>more</b>
90.852075	92.295317	it had
92.583965	95.542611	been very hard on Alice
95.542611	97.635312	<b>to to</b> live
97.635312	102.758821	in <b>la</b> lonely castle on so far <b>from city</b> from the city
103.047469	109.109085	tonight it was even ha harder <b>ba</b> because the storm fa fa fighting her
109.109085	111.201785	it not even
111.201785	115.820159	a good night for <b>ga gots</b> goat
115.820159	119.283940	<b>ga</b> ghosts
119.572588	121.015830	she thought
121.015830	123.830152	she put some
123.830152	126.355825	more firewood
126.355825	128.448526	into the <b>fa</b> fire
128.448526	132.706089	she knew she was only fooling
132.706089	134.005007	herself
134.654466	138.984191	<b>um</b> she knew the goat ghost
138.984191	141.870675	<b>wa</b> would be in the chapel again

Figure 4.4 Transcription example for speech sample after exporting the file using Audacity software. The red boxes have been added to highlight certain stuttering events by the author.

#### 4.4.1 Gold standard

The transcription guidelines follow the same conventions used for the orthographic transcription provided by UCLASS, but without capitalisation. Moreover, AMI transcription guidelines (Moore et al., 2005) have been applied for words unaffected by stuttering. In general, the two main rules in this process are:

- Full verbatim/time-aligned transcription. Write each spoken word as it is heard.
- To transcribe a sound repetition, an orthographic schwa sound is inserted to aid the pronunciation model in the ASR after each repeated sound, such as 'wa what'.

## 4.5 Annotation scheme

Word-level annotation is used as input to different machine-learning classifiers that detect stuttering events within a transcription. Transcriptions for this task were orthographic, not time-aligned, and included conventional forms to represent stuttering disfluencies, such as ('This is a a a amazing'). The annotation process includes the corresponding stuttering type

for each stuttered word and symbol 'NS' for each fluent word. Figure 4.5 presents a word level annotation example.

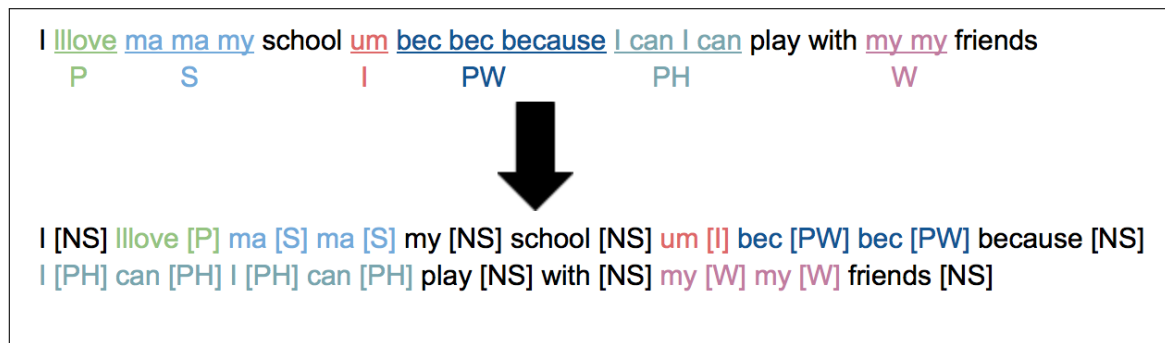


Figure 4.5 *Word-level annotation example.*

We provide word-level annotations for all orthographic transcriptions of Release Two, the 31 transcriptions by Howell of Release One and the 32 transcriptions by us of Release One, as described previously.

The stuttering events were identified and annotated with a special symbols. Yairi and Ambrose (2005) approach considered eight types of stuttering: 1) sound repetitions, which include phone repetitions of less than one syllable (e.g. 'fa face'); 2) part-word repetitions, which refer to a repetition of less than one word and one or more complete syllables (e.g. 'any anymore'); 3) word repetitions in which an entire word is repeated (e.g. 'mommy mommy'); 4) prolongations, which involve an inappropriate duration of a phoneme sound (e.g. 'mmmay'); 5) phrase repetitions that repeat at least two complete words (e.g. 'it is it is'); 6) interjections (this term is used in the stuttering literature; in ASR these are often known as 'fillers'), which involve the inclusion of meaningless words (e.g. 'ah,um'); 7) revisions that attempt to fix grammar or pronunciation mistakes (e.g. 'I ate; I prepared dinner'); and 8) blocking, which involves a stoppage of sound (any halting of speech, not just glottal stops) that can be momentary or longer and which occurs at an inappropriate place in an utterance, often including localised vocal tension. Each of the types of stuttering examined

in the annotation process is listed along with its corresponding abbreviations in Table 4.2. A UK-registered speech language therapist reviewed the annotation scheme.

Table 4.2 *Types of Stuttering.*

Label	Stuttering Type
I	Interjection
S	Sound repetitions
PW	Part-word repetitions
W	Word repetitions
PH	Phrase repetitions
P	Prolongation
NS	Non-stutter

#### 4.5.1 Gold standard

- Add [**W**] tag for any repeated words, such as 'my my'.
- Add [**S**] tag for any repeated sound and its affected word, such as 'ma ma my'.
- Add [**PW**] tag for any repeated part-word and its affected word, such as 'bec because'.
- Add [**PH**] tag for any repeated phrase, such as 'I can I can'.
- Add [**I**] tag for any interjected word, such as 'um, uh, ah'.
- Add [**P**] tag for any prolonged word, such as 'lllove'.
- Add [**NS**] tag for any non-stuttered word.

## 4.6 Agreement Measure: Cohen's kappa coefficient

Since the author conducted the transcription and annotation process, it is recommended that the processes be verified with evaluations conducted by external experts for a sample of the data. The sample should comprise at least 10% of all cases. Kappa is applied in the meta-analysis and content analysis domains when a researcher needs to determine the accuracy of coding for nominal variables based on raters' agreement. The Kappa statistics include classifying N items into C mutually exclusive categories; then, the agreement is calculated. There are variations in the way the kappa coefficient is calculated. For example, the difference in the number of raters and whether the distribution is fixed or marginal. Fixed marginal Kappa is used when the raters must distribute a specific number of items under each category; free marginal Kappa places no restriction on the number of cases placed into each category (Randolph, 2005). In the present study, the Kappa values range from -1 to 1, where -1 means 'complete disagreement' and 1 means 'perfect agreement'. The most common results for the Kappa values are shown in Table 4.3.

Values	Interpretation
< 0	Less than chance agreement
0.01–0.20	Slight agreement
0.21– 0.40	Fair agreement
0.41–0.60	Moderate agreement
0.61–0.80	Substantial agreement
0.81–0.99	Almost perfect agreement

Table 4.3 *Interpretation of Kappa (Randolph, 2005).*

We performed an inter annotator agreement analysis on 11% of the cases studied in this project. Two external raters worked on the task; this first was the author, and the second was



a certified speech language therapist. These raters compared their classifications, and the free marginal Kappa was applied, since there was no restriction on the number of cases under each category. Table 4.4 shows the Kappa statistics results and their interpretation.

Recording #	$\kappa$	Interpretation
1	0.5968	Moderate agreement
2	0.7681	Substantial agreement
3	0.6667	Moderate agreement
4	0.6005	Moderate agreement
5	0.5849	Moderate agreement
6	0.7667	Substantial agreement
7	0.287	Fair agreement
8	0.4915	Moderate agreement
9	0.3086	Fair agreement
10	0.6341	Substantial agreement
11	0.7188	Substantial agreement
12	0.5714	Moderate agreement
Average	0.5775	Moderate agreement

Table 4.4 *Kappa statistics and their interpretation.*

Most of the results were either 'Moderate agreement' or 'Substantial agreement' except sample number seven and nine which had 'Fair agreement'. In these samples the judgment was harder than others due to the high severity of stuttering. In general the results were acceptable as the average agreement was 'Moderate'.

## 4.7 Summary

There is a need to collect enough data on stuttered children's speech to build an ASR that can automatically recognise stuttering events within children's speech. The main challenge in this task is the limitation of available data. The only public corpus for children with stuttering in English is UCLASS, by Howell et al. (2009). The main challenge related to this corpus is that there is no transcription available for this corpus to train a suitable ASR. This chapter has discussed the first phase of building an ASR system for children's stuttered speech. It explained the transcription process using Howell's scheme guidelines combined with more general transcription guidelines from the AMI transcription guideline corpus (Moore et al., 2005). Depending on the difficulty of the transcription task and on the level of consistency and correctness one wants to achieve, the effort ranges are between 10–100-man hours for every 1 hour of recorded speech. However, it was reported that the impact of transcription errors on the performance of trained ASR systems is not as significant as one might expect. Sundaram and Picone (2004) reported that 16% of falsely labelled training data leads to a performance loss of 8.5% relative to the baseline on the Switchboard corpus. The initial transcriptions that we produced in this chapter had a fair number of transcription errors. Many of these errors occurred with names and places with which the transcriber (author) was unfamiliar. Because the task involved reading particular stories, the transcriber was able later to correct all transcription errors by listening to the same story from another recording sample. In addition, given the results of the average kappa agreement (moderate), the transcriptions contain a rather limited number of errors. Therefore, we are confident that transcription errors did not have a significant impact.

The second part of this chapter described the annotation process applied to each word to train machine learning classifiers to detect stuttering events among transcriptions produced by the ASR.

# Chapter 5

## Automatic Stuttering Recognition

The content of this chapter is based on the two conference papers previously published in (Alharbi, Simons, Brumfitt and Green, 2017; Alharbi et al., 2018).

### 5.1 Introduction

One of the main objectives of this thesis is to develop an automatic approach to help therapists tally the different classes of stuttering events and decide whether or not a case is a severe, borderline or normal disfluency case. Existing studies for automatic stuttering detection and classification can be grouped into two approaches: 1) Studies based on identifying stuttering events directly from speech signal, which we will henceforth call the '*direct identification approach*' (Howell et al., 1997; Chee et al., 2009; Świetlicka et al., 2013). 2) Studies based on analysing transcripts created by automatic speech recognition (ASR), which we will henceforth call '*the ASR approach*' (Nöth et al., 2000; Heeman et al., 2011, 2016).

The direct identification approach is the first approach developed to classify different stuttering events by class. However, there is a lack of sufficient data currently available to be able to generalise learning over stuttering events that are phonetically distinct to detect these events rather than classify them. An example of this is sound repetitions, which, from the point of view of counting stuttering events, could be any manner of sounds, acoustically

speaking, such as *ca*, *ba*, and *ma*. Because these sounds are all different, there is a need for significantly more training data that include them to be able to classify all of these sounds as one event (sound repetition).

Another issue has to do with detecting repetitions with different time spans such as sound, part-word, whole-word, or phrase repetitions and revisions. This gives rise to multiple types of repeating phenomena repeated over widely different scales, for example a phrase repetition in, '*this is cat this is cat*'. Generalising learning over different classes of stuttering events and making the system recognise different repetition classes over different time scales is a challenging task with the currently limited data.

A similar concept could be applied to prolongation events, prolongation representing any prolonged phone. Because prolonged phones - such as *sss*, *fff* and *zzz* - are different, more training data would be needed for identifying these occurrences as a single event directly from a speech signal. We conducted a pilot study to explore this by training a long-short-term memory (LSTM) to classify prolongation events. LSTM assumes the use of sequential data information. The aim of this pilot study is to observe the ability of a classifier in relation to a binary classification problem (prolongation/non-prolongation). This pilot study was conducted on manually segmented speech signals containing 120 prolongation segments and 16322 non-prolongation segments obtained from 48 speech recordings, UCLASS corpus/reading, Release Two. The results produced by the LSTM classifier were very poor, with only 10% precision and 4% recall achieved. The classifier faced two main challenges with the current data. The first was the small amount of training data. The second problem was the unbalanced data problem. In our current data, we have far more non-prolonged segments than prolonged ones. Therefore, we assumed that the different kinds of repetitions, including sound, word, part-word and phrase would be more harder with current challenges.

The adaptation of an ASR system is an alternative approach for which there is more general training data for children. Instead of detecting stuttering events, the ASR recognises speech and produces a full verbatim transcription, including different phonetic occurrences of stuttering. This allows stuttering events to be counted and provide a transcription for speech language therapist which is very helpful in the detailed diagnostic analysis.

This chapter demonstrates two ASR approaches for the automatic recognition of stuttered speech in children. The first uses an augmented LM approach, and the second is a task-oriented lattice. The crucial advantage of ASR-based approaches is that they provide textual information. Different phonetic stuttering events are then counted as automatic identifications from the transcript produced by the ASR. The results of the proposed ASR-based approaches are presented and discussed in Chapter 6.

The structure of this chapter is as follows: The proposed automatic speech recognition approaches are explained in section 5.2 where Section 5.2.1 reviews the previous work related to building ASR systems for children and the challenges related to this task. The speech corpus for the ASR used in these approaches is described in Section 5.2.2. Next, Sections 5.2.3 and 5.2.4 explain the LM augmentation and task-oriented lattice decoding approaches, respectively. Finally, a summary is presented in Section 5.3.

## **5.2 Automatic speech recognition (ASR) approaches for stuttering events detection**

Automatically detecting all the different types of stuttering events found in children's speech has been shown to be a significant challenge. The current study targeted the most common types of stuttering events for automatic detection, which included sound/part-word/word/phrase repetition as well as interjection and revision.

The following two sections will demonstrate the previous attempts to apply ASR on stuttering problem and the current data used in the proposed ASR approaches. Then, the two main approaches that were applied to detect all stuttering events except duration-based events, such as prolongation and blocks will be explained.

The first approach tried to detect stuttering events by augmenting the probability of stuttering events in the language model. The second approach applied a task-oriented lattice to build a more constrained and specified LM that allowed for a number of stuttering events.

### **5.2.1 Previous attempts to use ASR for stuttering**

Understanding children's speech is a well-known challenge, due to several factors, such as speech spontaneity, slow rates of speech and variability in vocal effort (Liao et al., 2015*b*). This increases the difficulty of automatically recognising children's speech, which has important applications in speech pathology and education. Children have smaller vocal tracts, which creates challenges in recognising their speech (Liao et al., 2015*b*; Russell and D'Arcy, 2007). Previous research has reported the poor performance of ASR systems when recognising children's speech (Liao et al., 2015*a*; Russell and D'Arcy, 2007). ASR systems trained on adults' speech may not perform well with children's speech due to the variation in the acoustic and linguistic characteristics of children's speech compared with adults' speech (Lee et al., 1999; Wilpon and Jacobsen, 1996; Potamianos, 1997; Claes et al., 1998). Great care needs to be taken to adapt or design models that target children's speech (Potamianos and Narayanan, 2003; Hämäläinen et al., 2014), such as age-matched training data (Elenius and Blomberg, 2004; Wilpon and Jacobsen, 1996). However, considerable variation in performance still exists between an ASR system trained and tested on adults' speech compared with an ASR system trained and tested on children's speech (Fringi et al., 2015).

Attempting to distinguish and detect stuttering events in children's speech adds to the complexity of this task. Available literature features ASR systems built for reading to study the possibility of applying ASR technology to recognise different types of stuttering disorders. It is easier for ASR systems to recognise read speech because it has prior knowledge of what the subject intends to say, which limits the search space for predicting the next word. Furthermore, the reading speech task is already applied as a part of most stuttering assessments (Gregory et al., 2003; Riley, 1994).

As mentioned in Chapter 3, Section 3.3.2, the first attempt at automating stuttering counts using speech recognition was provided by Nöth et al. (2000). Their system used a deterministic grammar for each phoneme because this was assumed to model the positional regularities of the different stuttering events during a reading task. However, the evaluation of the system did not report the correct recognised stuttering events or the false alarm rate.

Studies undertaken by Heeman et al. (2011, 2016) merged ASR outputs with the clinicians' manual annotations to produce corrected transcripts of stuttering speech. These studies aimed to help clinicians spend less time correcting transcriptions and more time analysing the clients' stuttering patterns. They observed that the quality of the word transcription could be improved by a relative factor of 7.5% by combining a clinicians' annotations.

In the current work, we are attempting to help therapists by providing them with an indication of the severity level of stuttering from an audio speech recording. The starting point for this is to adapt an ASR system to recognise stuttering events and provide a full-verbatim transcription. Then, the ASR output can automatically be processed to analyse the detected stuttering events to finally determine the severity level.

### **5.2.2 Stuttering dataset of children's read speech**

The current study is based on a standard reading task used by clinicians to diagnose stuttering in children. For training purposes, all the recordings of children's read speech were obtained

from the UCLASS Release Two (Howell et al., 2009). It contains 107 recordings of readings contributed by 40 different speakers.

A subset of 48 speech samples (totalling 120 minutes) were used in the current research. The 48 samples involved 25 recordings of ‘Arthur the rat’ and 14 recordings of ‘One more week to Easter’, with the rest containing other passages from books such as ‘Washington’ and ‘Alice fighting a ghost’. The reason for choosing this subset was to train the ASR system on the most frequent passages rather than confusing it with more than 50 recordings of different passages. Each speaker read a passage once. The passages were manually transcribed to serve as a reference, (See Chapter 4).

The data was divided into 40 samples, which were used as the training set, and four samples were used as a test set. As is usual with small datasets, a cross-validation (CV) technique was used to partition the stuttering data. A cross-validation technique is a sort of repeated training and evaluation process that is used when there is only a very little data. Therefore, there is a need to make the best use of the data in both training and testing. The way to apply this is to take a different portion of the whole data as a test data each time and training on the remaining data instead. In particular, CV was applied to the ‘Arthur the Rat’ passage given by (Abercrombie, 1964) because this passage was sampled most frequently and was chosen to evaluate the ASR system.

To improve the ASR system’s acoustic model, seven hours of recordings from the PF-Star (Russell, 2006) corpus of children’s read speech was added to the training set. The PF-Star corpus includes samples from 158 children aged between 4 and 14 years. Most of the children recorded 20 ‘SCRIBE’ sentences, a list of 40 isolated words, a list of 10 phonetically rich sentences, 20 generic phrases, an accent diagnostic passage (the ‘sailor passage’) and a list of 20 digit triples. This corpus contains simultaneous recordings from both a headset microphone and a desk microphone. Recordings from the desk microphone were used for training and testing to evaluate the results while taking into account the domestic acoustic



background. All the data preparation scripts were prepared to provide the necessary files for Kaldi toolkit (Povey et al., 2011). This toolkit is used mainly for speech recognition and it is written in C++. In this thesis, this toolkit is used to build both the acoustic model and to train an ASR system to recognise stuttering events. Table 5.1 provides a description and frequency of each stuttering event on 48 read recording samples obtained from UCLASS, Release Two.

Table 5.1 *Description and frequency of each stuttering class on UCLASS, Release Two, 48 read recording samples.*

<b>Stuttering class</b>	<b>Tag</b>	<b>Frequency</b>	<b>Description</b>
S	Sound repetition	2%	Sound repetition usually occur at the beginning of the word, multiple sounds can occur. For example: for prompt “ca ca come down just yet”
W	Word repetition	2%	Word repetition is simply a repeated word, multiple words can occur. For example: for prompt “come come down just yet”
R	Revision	1.8%	When a child trying to revise a sentence. For example: for prompt “He would answer he would only answer”
Ph	Phrase repetition	1.5%	When a child repeating a complete phrase. For example: for prompt “He would only he would only”
PW	Part-word repetition	0.33%	Repetition of a syllable or multiple syllable within a word, For example: for prompt “gar garden with an elm tree”
I	Interjection	0.22%	Occur when the child adding a sound or word such as um or ah

### 5.2.3 Augmentation of stuttering events in LM approach

Most existing ASR systems generate final word hypotheses based on the probability distribution of each word available in the training text corpora provided by the LM. The performance of the ASR system is highly dependent on the amount and style of the text seen in the training corpora. In general, rich text leads to a better model. However, the text used to train the model needs to match the language style used in the ASR system's application. Therefore, many problems have occurred in some ASR applications due to the difficulty of providing a good match, in-domain and sufficient text to reach a satisfactory level of performance. Several solutions have been proposed.

The first solution is LM adaptation. Many approaches to LM adaptation have been proposed, such as dynamic adaptation, which continuously updates the LM's probability distributions (Federico et al., 1999). Kawahara et al. (2008) studied the LM adaptation of automatic real-time lecture transcription using information provided in presentation slides used in a lecture. However, due to the small amount of text, and the fragmentary nature of the text content provided in the presentation slides, the author applied a global and local adaptation scheme. The use of adapted global topics is based on probabilistic latent semantic analysis (PLSA) by adding keywords shown in all the slides. For local adaptation, they also used a cache model to store each word mentioned in the slides used during each utterance, and the occurrence probabilities of these words were heightened, as they are more likely to be re-used. They reported that the proposed approach achieved a significant enhancement in recognition accuracy, particularly in the detection rate of content keywords.

Another possible solution is the application of data augmentation, which somehow generates extra artificial data for events that are not commonly observed in the available training data. The artificial data have to be generated from some other source of knowledge that provides some relative frequencies of the artificial events required to be generated.

Therefore, the revised probabilities in the LM correspond more closely to the target language required to be recognised.

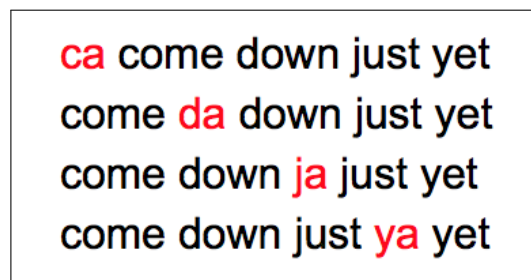
The method suggested in the current work began with the speculation that if the UCLASS read corpus (the corpus that includes stuttering events) is augmented with more artificial stuttering events, then the probability of these events will increase in general and will be considered during the recognition phase.

According to Yairi (1983), the most common types of disfluency are sound and word repetitions. We found empirically that words that a part of word and phrase repetition are more likely to be detected by a conventionally trained speech recogniser, whereas sound repetition are not recognised well in conventionally trained speech recogniser. Therefore, we focused our LM augmentation in sound repetition. Also, applying this to our dataset of stuttering speakers had the effect of increasing training occurrences in LM of other stuttering events which were there in the background of the speech that we augmented.

Moreover, sound repetitions will pose a challenge to the ASR system. The child will read a known passage in the test recording and might produce a sound repetition in any word of a given passage. Mostly, this sub-word (sound repetition) does not exist in the pronunciation dictionary or has a very low probability in the tri-gram LM. In addition, only adding those possible sub-words to the pronunciation dictionary will not solve this problem due to their low probability in the tri-gram LM. Therefore, as mentioned before, a proposed solution is to increase the probability of this event by adding more artificial sound repetitions to the UCLASS corpus to increase the chance of detecting sound repetition events.

The main approach of augmentation is to increase the occurrence of sound repetition in the LM. Each sound repetition is represented by a pseudo-word, which is mapped to a consonant + schwa, as described in Chapter 4. A stuttered corpus was applied to create the original tri-gram LM used to generate the additional sound repetition events. This would create a new augmented corpus with a greater proportion of stuttering events. Other

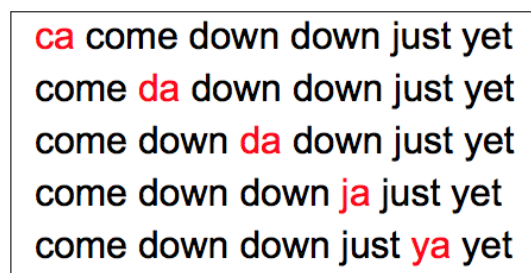
stuttering events also increased in frequency with the addition of repeated utterances with sound repetition.



ca come down just yet  
 come da down just yet  
 come down ja just yet  
 come down just ya yet

Figure 5.1 *Augmentation example.*

As demonstrated in Figure 5.1, for each utterance in the training corpus, such as ‘come down’, we automatically generated a stuttering event by taking the first phone of the first word and adding a ‘schwa’ sound after it; represented orthographically by ‘a’, as in ‘ca come’, and adding the whole utterance with the new stuttering event to the augmented corpus ‘ca come down’. Then, we took the first phone of the second word with a ‘schwa’ sound and added it as another new utterance, ‘come da down’.



ca come down down just yet  
 come da down down just yet  
 come down da down just yet  
 come down down ja just yet  
 come down down just ya yet

Figure 5.2 *Sound repetition augmentation example with already existing word repetition.*

Figure 5.2 shows one of the stuttering examples that were augmented and because it also had a word repetition, there were slightly more occurrences of those stuttering events in the training set as well. This example demonstrates how other stuttering events also increased in frequency with the addition of repeated utterances with sound repetition.

The algorithm used to build the augmented LM is described by Algorithm 1. It mainly added a sound repetition to each word per utterance, which increased the probability of

---

**Algorithm 1** Augmentation Algorithm

---

 $N$ : Number of utterances  $\rightarrow U_n$  $W_n$ : Number of words in  $U_n$  $S$ : stuttering event**for**  $W_1$  to  $W_n$  **do** $\hat{W}_n = S @ W_n$  $\hat{U}_n^w = \{ W_{n-1}, \hat{W}_n, W_{n+1} \}$ **end for**

---

stuttering in that word and, consequently, could lead to an improvement in the recognition of stuttering events overall.

In the current study, we trained an acoustic model using nine hours of speech, of which seven hours came from the PF-Star corpus (Russell, 2006) of normally fluent child speech and two hours initially from the UCLASS corpus. We needed to train on both datasets because training on just the UCLASS data alone was insufficient. In both corpora, children read from simple stories such as ‘Arthur the Rat’ and the ‘Poor Fisherman’. In the combined training set, the probability of stuttering events was small. Chapter 6 will demonstrate the results.

### 5.2.4 Task-oriented lattice approach

Given an approximate transcription (close to a manual transcript but not as exact), a more accurate transcription can be generated using a biased LM. This is known as a lightly supervised approach. It has been used successfully to generate improved transcriptions for acoustic model training, thus avoiding the need for expensive manual human transcription (Lamel et al., 2002; Chen et al., 2004; Chan and Woodland, 2004). It has also been used to align and correct approximate transcriptions of long audio recordings (Hazen, 2006) and for audio indexing and displaying subtitles. In a tutoring application, this approach was used by (Proença et al., 2017) to track and align a child’s read passage.

The current work used a specific lattice decoding approach to track and identify stuttering events in a reading task. We used a clean original prompt (OP) as an approximation of a manual transcript (that should include stuttering events). This step was applied to produce the best possible alignment for both clean words and stuttered words.

There were two main challenges. The first challenge came from all the extra words and sub-words that usually occur during the stuttering. The second challenge was how to increase the detection of the actual stuttering events by controlling the occurrence of the false alarm rate, which at this point, would affect the final determination of the stuttering severity level. The decoding approach must not be too unconstrained since this would increase the possibility of a false alarm. As a result, the intended decoding approach included grammar, which strictly followed the prompt, with the added options of sound/part-word/word and phrase repetitions, and revision.

For acoustic models, we obtained success in this task by applying a triphone hidden markov model of gaussian mixture models (HMM-GMMs), rather than neural networks for triphone decoding (HMM-DNNs). One explanation for this was that the total amount of training data (nine hours), including seven hours of PF-star corpus and two hours of UCLASS corpus, was not adequate for training HMM-DNN. Therefore, standard triphone HMM-GMMs were trained with the Kaldi toolkit (Povey et al., 2011).

At this stage, we present this approach to deal with stuttering events during decoding. The method followed two main steps:

1. An initial decoding run used the draft ASR hypothesis to automatically align with the OP by dynamic programming. This step eased the merging of several hypothesised segments into the corresponding OP utterance.
2. In the second stage, we used task-oriented finite state transducers (FST) for second pass decoding. All types of repetitions (sound/part-word/word/phrase) and revisions were targeted. These task-oriented lattices were automatically generated from the

OP, and weights were tuned to allow for possible stuttering events. A specific lattice was built to form an OP for each given utterance, in a way that allowed for stuttering events. This lattice and HMM models were used during this pass of the decoding. The FST was the lattice built for a specific utterance based on the word sequence of the OP. Additional elements were added to the lattice for each word, including an arc to go back after pronouncing a word. This allowed for word and phrase repetitions. In addition, a self-loop arc for each state allowed for sound and part-word repetitions. The weights for these arcs in the current study were empirically selected.

The designed approach, which allowed for sound repetitions (represented by the suffix `_S`), was based on stopping a word pronunciation at the first sound, such as ‘ga garden’. The part-word repetitions (represented by the suffix `_PW`) were allowed when the stopping occurred at the syllable boundary level, which is a frequent interruption point. For instance, for a word with four syllables [sy11.sy12.sy13.sy14], the allowed pronunciations for a part-word repetition could include any sub-sequence starting with the first syllable viz [sy11], [sy11.sy12] and [sy11.sy12.sy13]. To detect revision, we experimented to allow for a deletion option. Allowing for deletion produced worse alignment results compared with the results at the beginning of the study. This apparently occurred because the inclusion of ‘skip’ arcs in the lattices increased the freedom level, which allowed words to be matched incorrectly. Consequently, deletions were not allowed in this approach.

Scores: (#C #S #D #I) 6 0 1 0 REF: he would ALWA he would only answer HYP: he would **** he would only answer Eval: D <b>Revision</b>	Scores: (#C #S #D #I) 4 0 3 0 REF: HE WOULD ALWA he would only answer HYP: ** ***** **** he would only answer Eval: D D D <b>Revision</b>
Scores: (#C #S #D #I) 6 0 0 0 REF: come da down ja just yet HYP: come da down ja just yet Eval: <b>Sound repetition</b>	Scores: (#C #S #D #I) 4 0 3 0 REF: come DA down JA just yet HYP: come ** down ** just yet Eval: D D <b>Sound repetition</b>
Scores: (#C #S #D #I) 8 0 0 0 REF: and a garden with with an elm tree HYP: and a garden with with an elm tree Eval: <b>Word repetition</b>	Scores: (#C #S #D #I) 7 0 1 0 REF: and a garden WITH with an elm tree HYP: and a garden **** with an elm tree Eval: D <b>Word repetition</b>

**(a)** **(b)**

Figure 5.3 *Decoding*, (a) example showing the ability of the ASR to detect stuttering events after applying task-oriented lattices; (b) example showing the deletion of stuttering events using the baseline ASR.

Figure 5.3 shows the increased performance of the task-oriented over a baseline ASR. Figure 5.3 (b) demonstrates that there are certain stuttering events such as revision, sound repetition and word repetition where the baseline ASR simply deletes the stuttering events because it cannot cope with it. With our revised task-oriented lattice more word-level tokens are being recognised because of the retrained LM including revision, sound repetition and word repetition successfully. Even where the revised ASR failed to spot certain things such as the deletion in 5.3 (a), we were still able to detect a revision event.

Figure 5.4 demonstrates a lattice generated for the OP, ‘*garden with an elm tree*’. This can be considered as a forced alignment with additional features. A set of arcs were added to each word in the generated lattice. The first two arcs allowed for multiple occurrences of sound and part-word repetitions. Sound repetitions, represented by the suffix **\_S**, included all repeated sounds that could occur at the beginning of the word, such as *ga ga* in *garden*. Part-word repetitions, represented by the suffix **\_PW**, included all repeated syllables that could occur in the word *garden*. Traversing go-back arcs allowed for the possibility of both word and/or phrase repetitions and revisions, such as ‘*garden with garden with*’. All the results of this approach will be presented in Chapter 6.



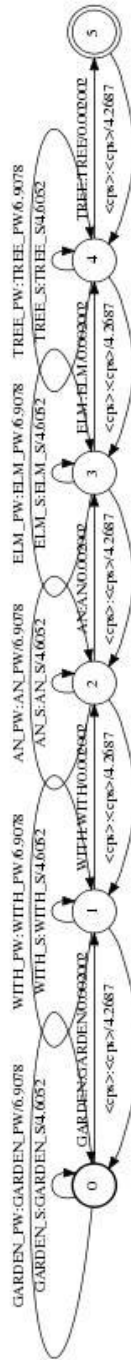


Figure 5.4 Corresponding FST graph for the prompt ‘GARDEN WITH AN ELM TREE’. The arc *GARDEN\_PW* allows part-word repetition and it could be [gar] or [den] while *GARDEN\_S* is sound repetition and it could be [ga]. The go-back transition allows word/phrase repetition and revision.

### **5.3 Summary**

Identifying stuttering events in children's speech is a difficult task due to a lack of available data needed for conventional statistical training methods. The rarity of particular stuttering events means that these would most likely be deleted in a transcript. We addressed this problem by proposing two main approaches. The first approach used a task-specific lattice re-scoring. The second approach relied on building an ASR system with an augmented LM to recognise stuttering in recorded audio files. Both approaches were specified for and applied to children reading stories to obtain orthographic transcriptions of what was said. This chapter explained the theory of these approaches, and Chapter 6 will demonstrate the experimental results for each approach and provide a comparison between them.

# Chapter 6

## Results of the Experiments with the Proposed ASR Approaches

The content of this chapter is based on the two conference papers previously published in (Alharbi, Simons, Brumfitt and Green, 2017; Alharbi et al., 2018).

### 6.1 Introduction

The previous chapter presented two automatic speech recognition (ASR) approaches for detecting most stuttering events. The first system examined the effect of augmenting a language model (LM) with artificially generated stuttering data, whilst the second system used a task-oriented lattices. This chapter presents the evaluation of the performance of the two proposed approaches by comparing how accurately they detected sound, word, part-word, and phrase repetitions, as well as revisions and interjections. This part of the evaluation concludes with an assessment of which approach is better. As time-based events, such as prolongation, cannot be detected, this type of stuttering event is addressed in Chapter7.

The chapter is structured as follows: Section 6.2 provides a summary of creating the baseline ASR system. The following sections present our experiments on the UCLASS data described in Section 5.2.2 for detecting stuttering events using the methods explained

in Section 5.2. A comparison between these two approaches is represented in Section 6.5. Finally, a summary is presented in Section 6.6.

The conventional *WordErrorRate* (WER) is used to measure the decoded output against an accurate manual transcription (including stuttering events). When performing a diagnosis task, the recall rate is sometimes reported, which is the percentage of detectable items that were actually found, and the difference from that in 100% is the miss rate or false negative rate (FNR). Similarly, classification systems sometimes falsely report detection of an event when that event does not actually occur, which is sometimes referred to as a false alarm, or false positive rate (FPR). The definitions of these metrics are as follows:

$$\text{Miss Rate (FNR)} = \frac{FN}{FN + TP}, \quad \text{FPR} = \frac{FP}{N},$$

where  $FN$  and  $TP$  refer to false negative and true positive,  $FP$  refers to false positive, and  $N$  refers to the number of original words, which include all false positives and true negatives. In the tables that follow, we will refer to these as misses and FPR.

## 6.2 Summary of building ASR-baseline process

This section presents a summary of the process that has been followed to create the current baseline ASR system.

### 6.2.1 Acoustic modeling

To build the acoustic model, and to develop a baseline ASR system that attempts to recognise stuttering events, the Kaldi ASR open-source toolkit was used (Povey et al., 2011). The Kaldi toolkit provides standard recipe scripts that were originally developed to run with different training datasets. These recipes enable Kaldi users to modify and manipulate the

scripts according to their requirements. Here, the Wall Street Journal (WSJ) recipe has been followed, beginning with the training of a monophone system that uses standard 13-dimensional Mel-frequency cepstral co-efficients with  $\Delta$  (deltas) and  $\Delta\Delta$  (accelerations). To reduce the channel effect, cepstral mean normalisation is applied. Then, using the information obtained from the monophone system, a triphone system is built using speaker-independent alignment. Next, a linear discriminant analysis (LDA) transformation is used to select the most discriminative dimensions from the larger context, and this includes taking five frames to the left and five frames to the right. A more refined step was developed using a maximum likelihood linear transform on top of the LDA feature and, once applied, the final GMM acoustic model was ready for use in offline decoding.

This particular acoustic model was trained with nine hours of the PF-STAR corpus of British English children's speech as the training subset (Russell, 2006). In addition, reading speech samples obtained from Release 2 UCLASS (Howell et al., 2009) were used to retrain the acoustic model to improve the ASR's ability to recognise stuttering speech.

### 6.2.2 Language modelling and pronunciation dictionary

The Kaldi toolkit does not provide any language model tools, although some example links were added to help the developer identify and use the desired language model. Here, the SRI Language Modelling (SRILM) n-gram toolkit was used because of its significant size and because it covers the language in the current model's training data (Stolcke et al., 2002). To create an effective language model, the text corpus and the word list of the entire training dataset must be used; an n-gram of this dataset must then be generated as input for the Kaldi kit. Here, two statistical trigram language models using the SRILM tool were built to specify how likely a word is to follow any other word in the audio file. The first language model is used for the PF-STAR data while the second trigram model includes story text read in stuttering speech. In this model, each word is interpreted as a different word according to

its position; the trigram then forces the ASR system to recognise the words in the order that they occur by placing all the probability for a word on the word that follows it in the story.

The language model creation process also requires a pronunciation dictionary that includes the phonetic sequence(s) for each word; the British English Example Pronunciation (BEEP) dictionary was here used for this purpose (Robinson, 1996). For words that do not appear in the dictionary, such as the stuttered words, the Sequitur tool was applied to estimate the phonetic sequence given the letters of the word (Bisani and Ney, 2008). This provides an estimate of the pronunciation of the unavailable word which it is then possible to manually check to ensure they are correct.

### 6.2.3 Pilot study

The Kaldi ASR setup was twice evaluated with existing corpora. In the first baseline experiment, the GMM acoustic model was trained on the PF-STAR corpus and evaluated using a test set previously designed for the purpose. The model was also evaluated using the stuttering test set to assess the algorithm's performance under stuttering (and noise) conditions. Table 6.1 presents the word error rate (WER) results obtained from using PF-STAR data only to train the ASR system.

Table 6.1 *WER (%) using PF-star as a training set.*

<b>Algorithm</b>	<b>Test Set</b>	<b>% WER</b>
GMM	PF-star	16.9
GMM	sub-UCLASS	84.74

The performance of this ASR with the PF-STAR test set is relatively good at 16.9%. However, and as expected, assessment of sub-UCLASS stuttering data by an acoustic model that is only trained on the PF-STAR corpus causes its performance to decline. The decline in performance is primarily due to the high levels of noise as well as the stuttering events that

the system has not been trained to interpret. Moreover, the PF-STAR-trained model had not seen most of the vocabulary in the stuttering set.

In the second experiment, the GMM acoustic model was trained on the PF-STAR corpus and the stuttering UCLASS samples. The obtained model was again evaluated using the test set designed for the PF-STAR data as well as the stuttering test set to measure algorithm performance under stuttering (and noise) conditions. Table 6.2 presents the WER results from using the full set of training data.

Table 6.2 (%) using PF-star with stuttering data as a training set.

Algorithm	Test Set	% WER
GMM	PF-star	16.58
GMM	sub-UCLASS	8.41

After adding two hours of stuttering data to the training process, the decoding performance is enhanced to 8.41%. with the UCLASS set. This improved result is expected as all of the story vocabulary already exists in exact order in the second language model (built from PF-STAR and UCLASS set) which sufficiently trains the ASR to detect all of the words. However, the current model fails to detect stuttering events due to the scarcity of such events in the applied language model. Therefore, the experiments regarding the two proposed approaches are described in Sections 6.3 and 6.4.

## 6.3 Augmentation of stuttering events in LM approach

### 6.3.1 Acoustic modeling

The acoustic model here applied is similar to that derived from the WSJ recipe and described in Section 6.2.1 The experiments relating to augmenting stuttering events in the language model use two ASR systems, and the same acoustic model is used across both.

A large seven-hour PF-STAR database was used to train the acoustic model in speech, and a small two-hour UCLASS dataset was also used to train both the acoustic and language models. The applied method is to use the PF-star data for training purposes, but to split the UCLASS data, which contained a mixture of fluent and stuttering speech as the test data. As a result, we made a variation on the cross-validation (CV) training approach, which reserves one part of the UCLASS data for testing but combines the remainder with the training data.

The main passage used to test the ASR system in both approaches (augmentation of stuttering events in LM approach and task oriented lattice approach) was ‘Arthur the Rat’, because it was the passage that appeared most frequently in the UCLASS data. These experiments were performed using six-fold CV sets to verify the reliability of the model’s performance tested on a total of 25 stuttering recordings of the ‘Arthur the Rat’ passage. The six-fold CV sets were determined after dividing the 24 recordings of the ‘Arthur the Rat’ passage by four, and the remaining recording was added to the last partition. Thus, each partition included four speech recordings except for the last partition, which included five recordings representing 16% of the complete stuttering data (25 recordings).

Table 6.3 *Summary of training and test data.*

<b>Set</b>	<b>Database</b>	<b>Speaker overlap</b>	<b>Duration</b>
Train	UCLASS/PF-star	No speaker overlap	Nine hours, including seven hours of PF-star corpus and two hours of UCLASS corpus.
Test	UCLASS	No speaker overlap	ASR model’s performance tested on a total of 25 stuttering recordings of the ‘Arthur the Rat’ passage. The duration of each recording is about 3 minutes.



### 6.3.2 LM and pronunciation dictionary

The SRILM toolkit (Stolcke et al., 2002a) was used to create the language models in this study. In this process, the text corpus and word list of the training dataset is used. Then, an n-gram can be generated as input for the Kaldi kit. For this, the SRILM tool is used to build trigram language models.

As outlined above, the experiments in this approach use two ASR systems:

1. **First ASR system:** The first system uses a trigram language model trained on UCLASS data only; this model is used for the baseline experiment. The corpus here essentially comprises the training data text which contains the training set of the UCLASS set. This experiment is designed to examine the baseline ASR's ability to identify words in stuttering events when trained on a language model that only contains a few stuttering events and therefore possesses low probability to recognise these words.
2. **Second ASR system:** In the second ASR system, two different LMs were used to build the model via the linear interpolation technique and to create a lexicon that is suitable for the newly-generated stuttered corpus. Notably, all LMs were implemented using Kneser-Ney discounting and standard back-off, and they were trained with the SRILM toolkit (Stolcke, 2002). The LMs are described in greater detail below.
  - $LM_1$ : The first language model is built using the UCLASS training set corpus; it is exactly the same as that created for the first ASR system above;
  - $LM_2$ : The second language model is an augmented tri-gram LM, which was trained on the augmented data that were generated using the approach described in Section 5.2.3. The goal of augmentation is to increase the occurrence of sound repetition in the corpus that is used to train the LM. Each sound repetition is represented by a pseudo-word, which is mapped to a consonant + schwa. An

'utterance' is a short word-sequence delimited by natural pauses. The training corpus is designed as a collection of utterances. If every word in an utterance is augmented for sound-repetition, then multiple copies of that utterance will be added to the corpus. As a result, the frequency of other stuttering phenomena reaching across many words was also increased in the LM.

- $LM_1+LM_2$ : This is the combined version of the previously developed two LMs: LM1 and LM2. This was done by interpolating these two LMs using the same word list, which includes additional stuttering words. Interpolation weights were optimised and tuned using maximum likelihood optimisation on the dev set, which includes 4 samples from different stuttering recordings that formulate about 12 mins of speech. Importantly, all three developed LMs ( $LM_1$ ,  $LM_2$  and  $LM_1+LM_2$ ) were considered in the recognition process.

The BEEP dictionary is used to create a pronunciation dictionary that consists of the phonetic sequence(s) for each word (Robinson, 1996). For words that are not in the dictionary, such as the stuttered words, the Sequitur tool is applied to estimate the phonetic sequences when given the letters of a word (Bisani and Ney, 2008). This provides an estimate of the pronunciation of an unavailable word, a process that, for some stuttering vocabularies, is also manually checked to ensure they are correct.

### 6.3.3 Experiments

#### Baseline

Baseline experiments are conducted to determine how the ASR behaves when trained on mostly fluent speech (from PF-Star) with only a small amount of stuttering data (from UCLASS). As mentioned previously, the baseline ASR acoustic model is a model trained on the PF-star corpus with the UCLASS corpus. The UCLASS corpus transcriptions include

stuttering events. The LM of the baseline ASR depends on the text corpora available for the training data, which belong to the UCLASS corpus.

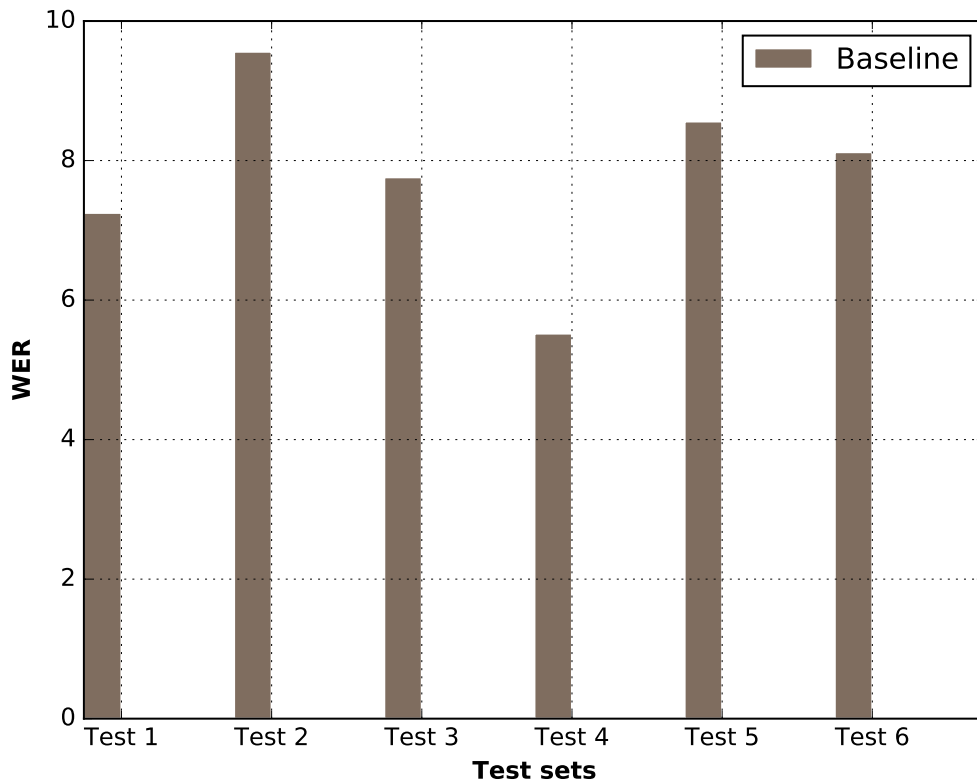


Figure 6.1 *WER results of the LM built from training set built from the training set.*

Figure 6.1 demonstrates the WER achieved in the baseline experiments. The system performs well in most test sets, with an average of 7.8% WER. This is primarily because the task is a reading task, and most of the words' contexts already exist in the LM. Therefore, the system can easily recognise fluent speech and tends to remove stuttered words.

Table 6.4 *Baseline experiment results when trained on the LM built from stuttered training data. 'n/a' means that this event was not present in the test set.*

Measures	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR
Stuttering type	S		W		R		Ph		PW		I	
Baseline Test1	90%	0%	40%	0.29%	50%	0%	n/a	n/a	n/a	n/a	n/a	n/a
Baseline Test2	62%	1.1%	60%	0.82%	67%	0%	60%	0%	n/a	n/a	100%	0.82%
Baseline Test3	100%	0.54%	67%	0.82%	80%	0%	60%	0%	n/a	n/a	100%	0%
Baseline Test4	100%	0.27%	43%	0%	50%	0.27%	80%	0%	n/a	n/a	100%	0%
Baseline Test5	80%	0.54%	80%	0%	100%	0%	67%	0%	100%	0%	100%	0%
Baseline Test6	88%	0.82%	50%	0.27%	40%	0%	100%	0%	n/a	n/a	100%	0%
<b>Average</b>	86.9%	0.55%	56.7%	0.37%	64.5%	0.05%	61.2%	0%	16.7%	0%	83.3%	0.14%

For a diagnosis system that determines whether patients should start receiving a treatment, false negatives (FNs) (or misses) are more important than false positive (FPs) because if the FN is high, this means that the system fails to diagnose patients who genuinely require treatment (Lever et al., 2016; Powers, 2011). Table 6.4 demonstrates the results of evaluating the performance of the baseline ASR. The results clearly suggest that the model's performance is very poor in terms of FN, as the miss rate is very high.

As clearly shown in table 6.4, the system misses all interjection events and only detects a few sound repetition events in the current training set due to rarity of these events. There are only two part-word events in *Test5*, and they are also missed by the baseline system. The current baseline model can only detect 29% of the total number of stuttering events, which is not sufficient for helping a therapist diagnose a child who stutters. The poor performance of the ASR system in detecting stuttering events can be explained by the high perplexity of the LM; the probability of describing stuttering events in the corpus is small because of the limited data available.

In the baseline experiments, we also find that the generated pronunciation in the pronunciation dictionary for some stuttering events is faulty. For example, incomplete words

produced by sound or part-word repetition are considered as an out of vocabulary (OOV) by the Sequitur tool. This penalises both the LM and the WER, and had to be fixed manually in the pronunciation dictionary.

### Augmentation

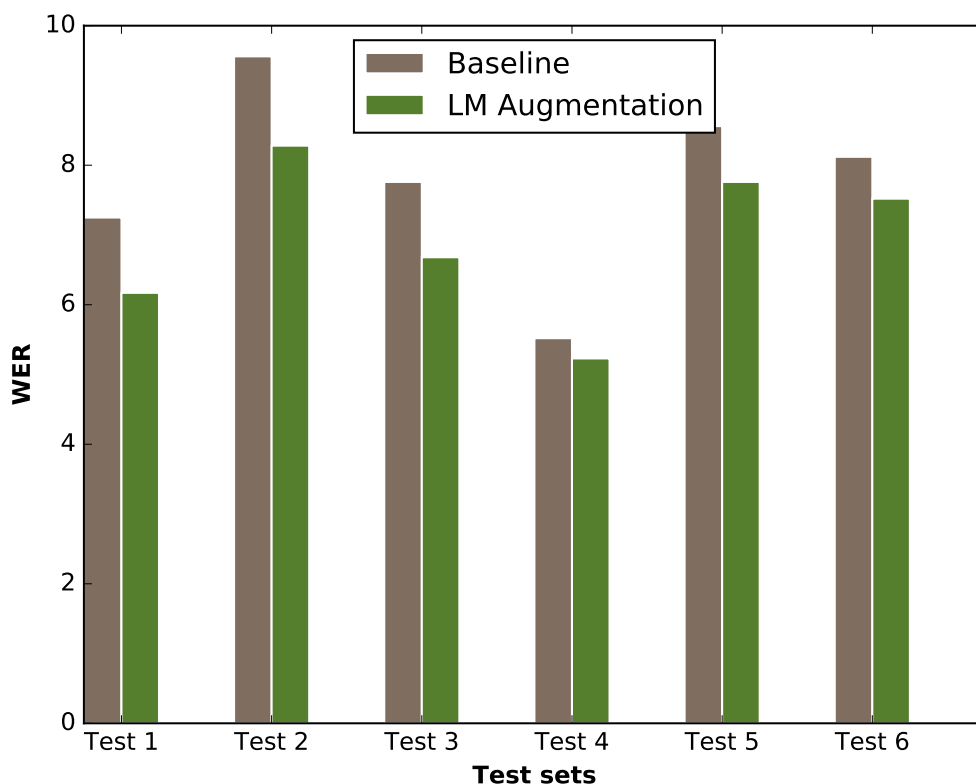


Figure 6.2 WER improvement between the LM built from the training set and LM built from the augmented corpus with artificial stuttering data.

After creating an LM with an augmented corpus, as explained in Section 5.2.3, we repeated the training with the augmented LM and analysed the impact on ASR performance. Figure 6.2 demonstrates the WER improvement between the LM built from the training set and the LM built from the augmented corpus with artificial stuttering data. The WER of the ASR is improved for all CV test sets with an average of 12% improvement, relative to the

baseline. One explanation might be that the improvement results from the correction of the pronunciation of stuttering events in the pronunciation dictionary, which positively affects both the LM and the WER and detects more stuttering events in the CV test sets.

```
id: (m_0048_11y1m_2-108.030098-110.903239)
Scores: (#C #S #D #I) 5 1 0 0
REF: well said the old RAIN angrily
HYP: well said the old RAT angrily
Eval: S

id: (m_0053_11y1m_1-108.939683-113.921681)
Scores: (#C #S #D #I) 10 0 0 1
REF: the idea of *** immediate decision was too much for him
HYP: the idea of THE immediate decision was too much for him
Eval: I

id: (m_0064_12y2m_1-11.74749-14.595367)
Scores: (#C #S #D #I) 9 1 0 0
REF: he WOULD say yes and he wouldn't say no either
HYP: he WOULDN'T say yes and he wouldn't say no either
Eval: S
```

Figure 6.3 Example of insertions and substitutions that affect the WER but do not affect the stuttering analysis process.

As we obtained a slight improvement in the WER, there is a need to consider whether the WER is a particularly good measure of an improvement that will help with stuttering event detection. When we look at examples of word substitutions and insertions presented in Figure 6.3, it is clear that these affect the WER, but this happens independently of the existence of stuttering events. Therefore, the WER may not be a particularly good statistic for detecting stuttering, because what we actually want to do is judge our ASR approaches at the end with the missed stuttering events in the transcriptions.

Table 6.5 Miss rate and false positive rate results after applying the augmented LM approach, 'n/a' means that this event was not present in the test set.

Measures	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR
Stuttering type	S		W		R		Ph		PW		I	
Baseline Test1	90%	0%	40%	0.29%	50%	0%	n/a	n/a	n/a	n/a	n/a	n/a
LM-AUG Test1	85%	1.0%	33%	0.55%	40%	0%	n/a	n/a	n/a	n/a	n/a	n/a
Baseline Test2	62%	1.1%	60%	0.82%	67%	0%	60%	0%	n/a	n/a	100%	0.82%
LM-AUG Test2	46%	2.7%	40%	0.55%	67%	0%	40%	0%	n/a	n/a	75%	0.27%
Baseline Test3	100%	0.54%	67%	0.82%	80%	0%	60%	0%	n/a	n/a	100%	0%
LM-AUG Test3	64%	1.0%	33%	0.27%	20%	0%	20%	0%	n/a	n/a	0%	0%
Baseline Test4	100%	0.27%	43%	0%	50%	0.27%	50%	0%	n/a	n/a	100%	0%
LM-AUG Test4	60%	0.55%	43%	0%	0%	0%	50%	0%	n/a	n/a	100%	0%
Baseline Test5	80%	0.54%	80%	0%	100%	0%	67%	0%	100%	0%	100%	0%
LM-AUG Test5	40%	2%	60%	0.82%	0%	0%	33%	0%	100%	0%	67%	0%
Baseline Test6	88%	0.82%	50%	0.27%	40%	0%	50%	0%	n/a	n/a	100%	0%
LM-AUG Test6	75%	1.3%	40%	1.1%	40%	0%	50%	0%	n/a	n/a	100%	0%

Table 6.5 shows the performance obtained with the CV of the ASR output and illustrates the detailed percentages of stuttering events detected compared to the ground truth stuttering events.

The detection of the sound, word, and phrase repetitions and revisions improved due to the probability of these events increasing after augmenting the sound repetition in the LM, as demonstrated by the lower miss rate score.

If the reader refers back to the algorithm used for LM augmentation in Chapter 5, Section 5.2.3, it can be seen that our strategy of repeating certain utterances with sound repetitions applied to different words in these utterances would also increase the LM frequency of pairs of repeated words or even repeated phrases. Hence, our LM augmentation also improved the probability of recognising word and phrase repetition. One observation is that there are no changes in recognising phrase repetitions in *Test4* and *Test6*. There are two repeated phrases

in *Test4* and two repeated phrases in *Test6*, but the ASR only recognised one out of the two in both test sets. This miss might be due to recording quality.

As mentioned previously, augmentation implicitly increases the probability of all stuttering events. Therefore, the probability of recognising these events increases. Part-word (PW) repetition events occurred only eight times across the stuttering data set, representing only 2% of other stuttering events, and they occurred only two times in *Test5*. All eight occurrences of PW repetition occurred chiefly in compound words, which are formed by combining two words, such as ‘anywhere’ or ‘anymore’. For example, a child who stutters is more likely to stutter on a compound word than a non-compound word and to say ‘any anymore’, which is considered a PW event. Therefore, a PW event that is a non-compound word, such as ‘Ar-Arthur’ or ‘fall-falling’, was missed in the test set by the ASR. The recognition of these events cannot be improved by the augmentation process, as they are not seen in the training data. The augmentation model was not designed to create PW instances due to the low incidence of PW events in the applied test set.

Table 6.6 *Average miss rate and false positive rate results.*

Test folds	Baseline		LM augmentation	
	misses	FPR	misses	FPR
Test1	60%	0.10%	52.66%	0.51%
Test2	63%	0.55%	46.86%	0.70%
Test3	81.4%	0.27%	27.7%	0.25%
Test4	68.6%	0.11%	50.6%	0.11%
Test5	87.8%	0.10%	50%	0.47%
Test6	65.6%	0.22%	61%	0.48%
Average	71%	0.22%	48.13%	0.42%



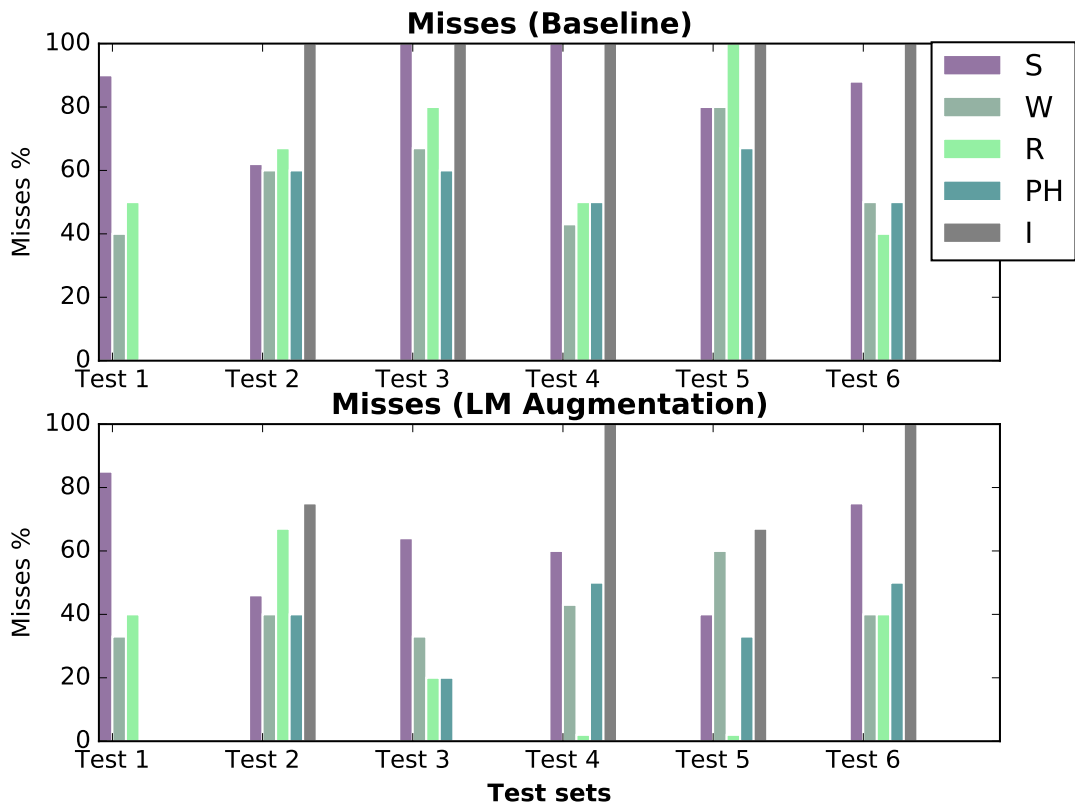


Figure 6.4 This figure represents the improvement in the miss rate after applying LM augmentation.

As illustrated in Figure 6.4, there is a clear improvement in the results compared to the baseline experiments. The average miss rate improvement of all stuttering events is 32.21% relative to the baseline. On the other hand, the average false alarm rate increases to reach 0.42%.

After analysing the results in more detail, it becomes clear that most of the false alarms for sound repetitions occur in same positions of prolongation and blocks events, which might increase the number of false alarms in a test set. Another observation is that the baseline cannot detect any interjections and produces a number of false alarms in *Test2*, while the results slightly improve after augmentation (25% improvement) in that set. The miss rate improved dramatically in some cases, most often because the misses were related to very

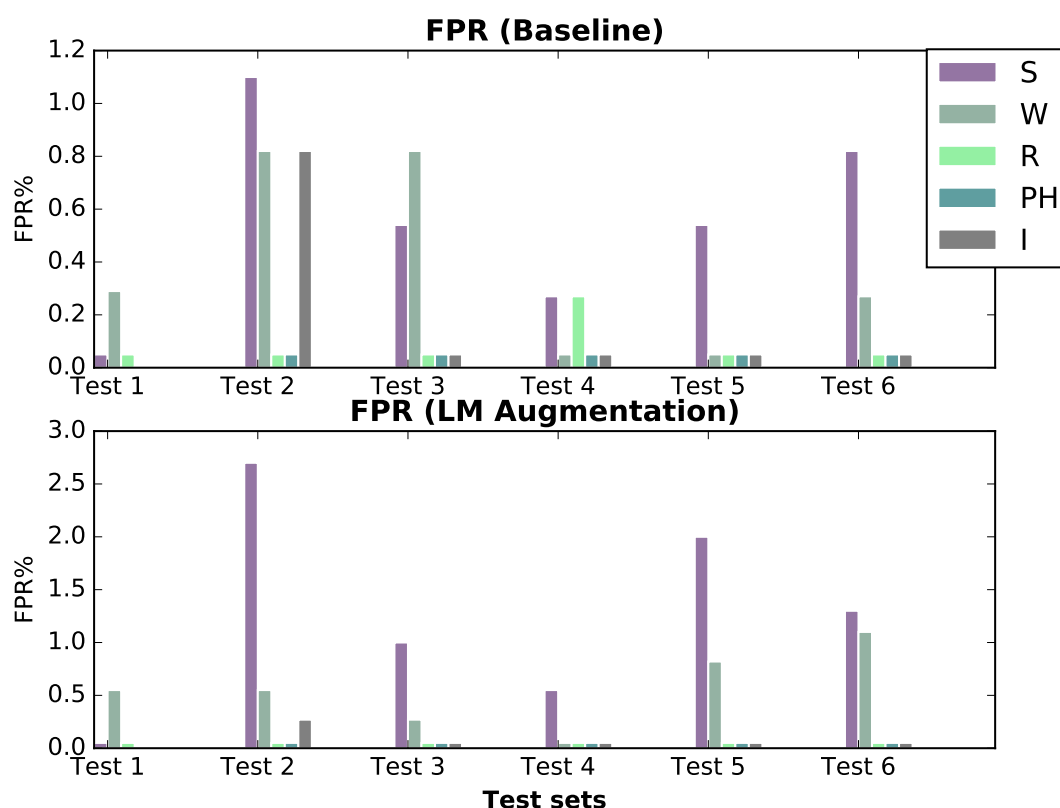


Figure 6.5 This figure represents the false alarm rate increase after applying LM augmentation.

few occurrences of these kinds of stuttering events in the test data. For example, in *Test3*, there was only one interjection that was missed in the baseline experiment and detected after augmenting the LM, which improves the miss rate to 0%.

In revision and phrase repetition types, there are no false alarms produced, except for one case in *Test4* that occurred because of an insertion between two repeated words, producing a revision false alarm.

Given that the miss rate decreased from an average of 71% to 48% and the FPR increased to 0.42%, which is still small, there is potential room for improvement in this approach. There is a possibility of augmenting the stuttering events that were least recognised, such as interjection events, to increase the ability of the LM to better recognise them, as these

events were least recognised across all test sets. However, this would involve increasing the frequency of occurrences of this event in the LM, which also might cause the FPR to increase.

## 6.4 Task-oriented lattice decoding approach

As mentioned in Chapter 5, Section 5.2.4, task-oriented lattices are automatically generated from the original prompt (OP) for each given utterance in a way that allows for stuttering events. All types of repetitions (sound/part-word/word/phrase) and revisions are targeted. The FST is the lattice built for a specific utterance based on the word sequence of the OP. Additional elements are added to the lattice for each word, including an arc to go back after pronouncing a word. This allows for word and phrase repetitions. In addition, a self-loop arc for each state allows for sound and part-word repetitions. The weights for these arcs in the current study are empirically selected and tuned to allow for possible stuttering events.

The baseline for this experiment is an ASR with a deterministic LM. This baseline is different from the baseline of the previous experiment, which was based on a statistical LM. This deterministic LM baseline is applied to be comparable with the advanced LM build from task-oriented lattice. The deterministic LM is simply creating a LM from the original prompt.

A detection error trade-off (DET) curve, as presented in Figure 6.6 can be obtained by applying a large search beam while decoding and varying the lattice rescoring weights and the word insertion penalty.

The following sub-sections discuss the results obtained from the baseline experiments and after applying the proposed task-oriented lattice approach on the same test sets to which the LM augmentation approach is applied. These experiments are performed using six-fold CV sets to verify the reliability of the model's performance. The six-fold CV sets are determined after dividing the 24 stuttering recordings of the 'Arthur the rat' passage by four.

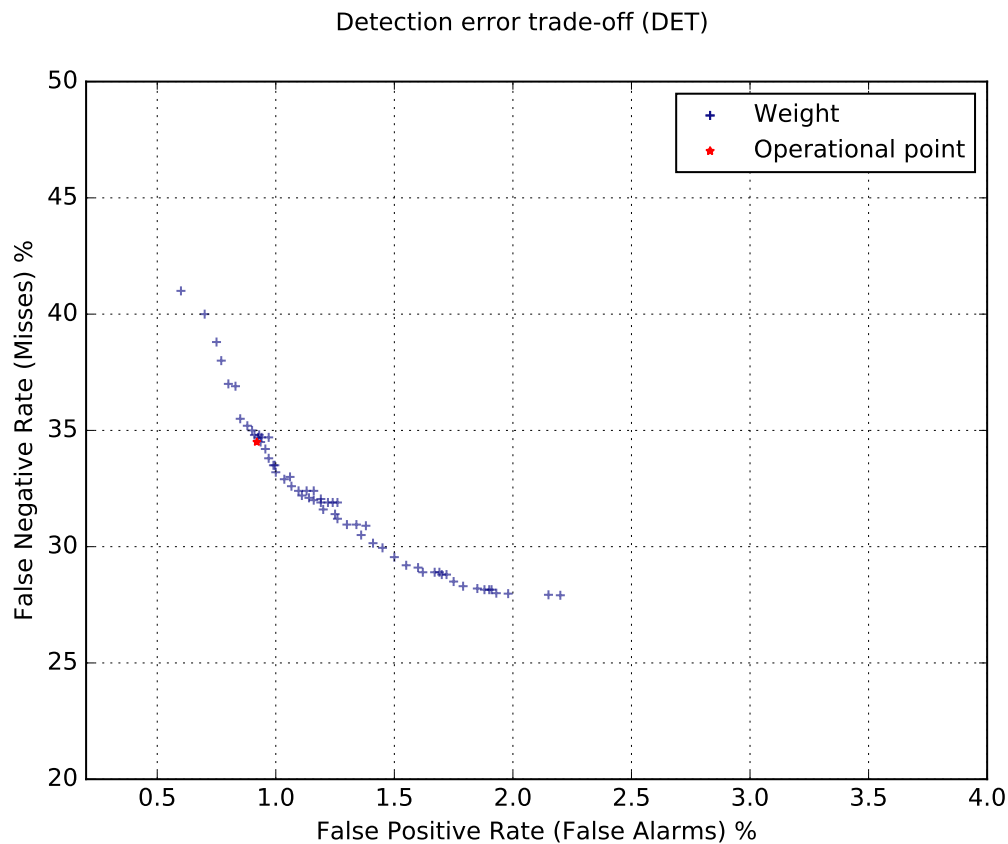


Figure 6.6 *Detection error trade-off (DET) for the detection of sound/word/phrase repetitions and revisions in the development set for the task-oriented decoding approach. The optimal point is reached when the best weights are applied.*

## 6.4.1 Experiments

### Baseline

The first experiment evaluates the performance of the ASR when the LM relies on the OP only. We evaluate the ASR's performance when applying an LM built from the OP (deterministic LM). The lattices are rescored using a deterministic LM created from the OP with no additional features. The obtained WER is demonstrated in Figure 6.7. As expected, the results confirm that the ASR, when built from the OP, tends to delete all stuttering events. Thus, the miss rate is 100%.

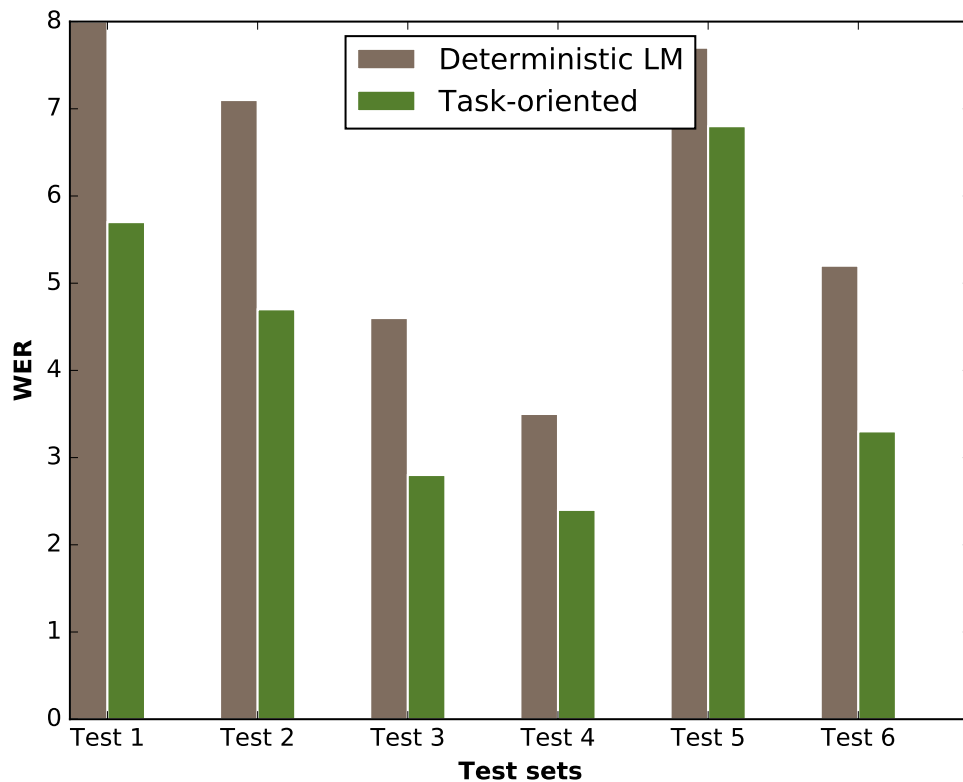


Figure 6.7 WER improvement between the deterministic LM built from the original prompt and the LM built from the task-oriented lattices.

We compare between the two baseline models to see how they performed. It is clear that the deterministic LM is not capable of detecting any stuttering events, because it merely expects one word to proceed to the next. On the other hand, the baseline statistical model can detect some stuttering events, such as a whole word or a higher number of repetitions, because of the slight probability that certain things could be repeated. The WER for the deterministic LM is better than the baseline statistical LM because there is always a small probability of a different sequence of words in the statistical model. However, there is no such possibility in the deterministic LM.

### Decoding with task-oriented lattices

The WER can be analysed as an initial measure of the deterministic LM and the task-oriented lattice with additional transitions. Figure 6.7 show that the WER improve for the task-oriented lattices because the test data contains stuttering events and the deterministic lattice is unable to detect any stuttering events at all. Therefore, there will be sound and words repetitions that have been picked up in the task-oriented lattices and not have been picked up by the deterministic one, which gives rise to the WER improvement. Although the WER improves that might not improve the best overall results, as the focus is on detecting particular stuttering events measure.

To evaluate the performance of the ASR system in detecting stuttering events, we use similar criteria to those applied in the NIST scoring tool (Fiscus et al., 2007): insertions of stuttering events are considered false alarms, deletions are considered misses and the substitution of detected events by events from other stuttering categories are considered misses.

Table 6.7 Miss rate and false positive rate results after applying the task-oriented lattice approach. 'n/a' means that this event was not present in the test set.

Measures	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR
Stuttering type	S		W		R		Ph		PW	
Task-oriented-lattices Test 1	60%	0.54%	0%	0.29%	25%	0%	n/a	n/a	n/a	n/a
Task-oriented-lattices Test 2	38%	1.64%	27%	1.1%	33.3%	0%	25%	0%	n/a	n/a
Task-oriented-lattices Test 3	54%	0.54%	33%	0.27%	20%	0%	16%	0%	n/a	n/a
Task-oriented-lattices Test 4	40%	0.27%	43%	0.27%	0%	0%	50%	0%	n/a	n/a
Task-oriented-lattices Test 5	25%	1.4%	20%	0.54%	0%	0%	33%	0%	50%	0%
Task-oriented-lattices Test 6	63%	0.27%	25%	0.82%	40%	0%	50%	0%	n/a	n/a

Table 6.7 presents the results after decoding using task-oriented lattices. The least-well-detected events are sound repetitions, for which the miss rate reaches 60% in *Test1* and 63% in *Test6*. A detailed investigation shows that these deleted sounds come from low-quality

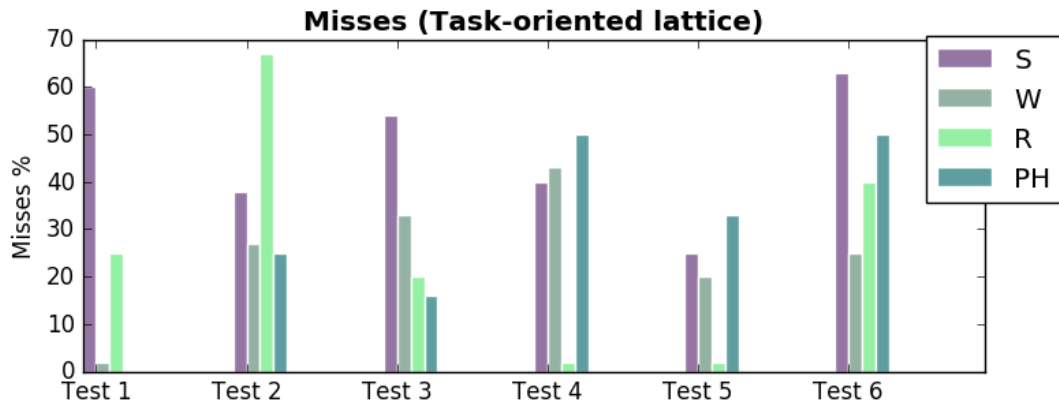


Figure 6.8 This figure represents the misses after applying Task-oriented lattice.

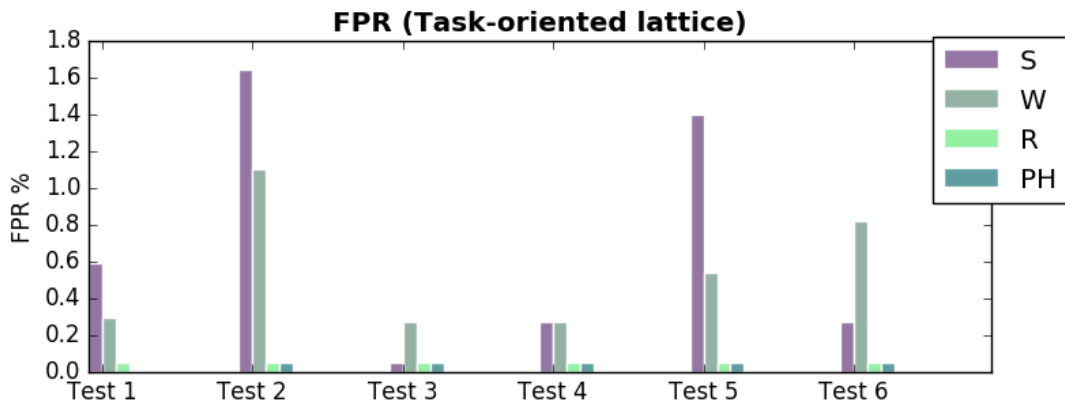


Figure 6.9 This figure represents the false alarm after applying Task-oriented lattice.

recordings. Additionally, these sounds are judged to be barely recognisable, even by humans. The miss rate for word repetitions reaches an average of 24.6% with an average of 0.54% false alarms.

One observation is that there are no false alarms produced in revision and phrase repetition types and that there are more false alarms produced by sounds and words. This might be because the weights assigned to the back-arcs is lower than the weights assigned to the self-loop arcs. As mentioned before, the probability of part-word repetition events is only 2% in the training set and only occurs in *Test5*. The ASR is able to detect 50% of these events with no false alarms.

Table 6.8 *Average miss rate and false positive rate results.*

Test folds	Task-oriented lattice	
	<b>misses</b>	<b>FPR</b>
Test 1	28.3%	0.28%
Test 2	39.25%	0.68%
Test 3	30.75%	0.20%
Test 4	33.25%	0.14%
Test 5	25.6%	0.38%
Test 6	44.5%	0.27%
Average	33.60%	0.32%

Using the re-scoring approach, we preserve an average of 66.4% of stuttering events. By contrast, the average false alarm rate is 0.32%, which is considered relatively low. This is mainly due to the constraints applied in each task-specific lattice. Given that miss rate decreases this much and the FPR is quite low, we believe that there is a possibility of improvement in this approach. However, the main problem we faced was during the tuning weights process of each arc. For example, when we made sound repetition more probable, the miss rate for the word and phrase repetitions increased again. Therefore, we reverted to the red point shown in the curve shown in Figure 6.6. Empirically, it did not converge any further than this because, when we tried to increase the likelihood of one kind of stuttering event, that improved the overall miss rate. However, at the same time, the miss rate for the corresponding ones deteriorated significantly. Thus, we achieved approximately the same or worse results in terms of overall misses, for a rather low false positive rate.



## 6.5 Comparison between two proposed approaches

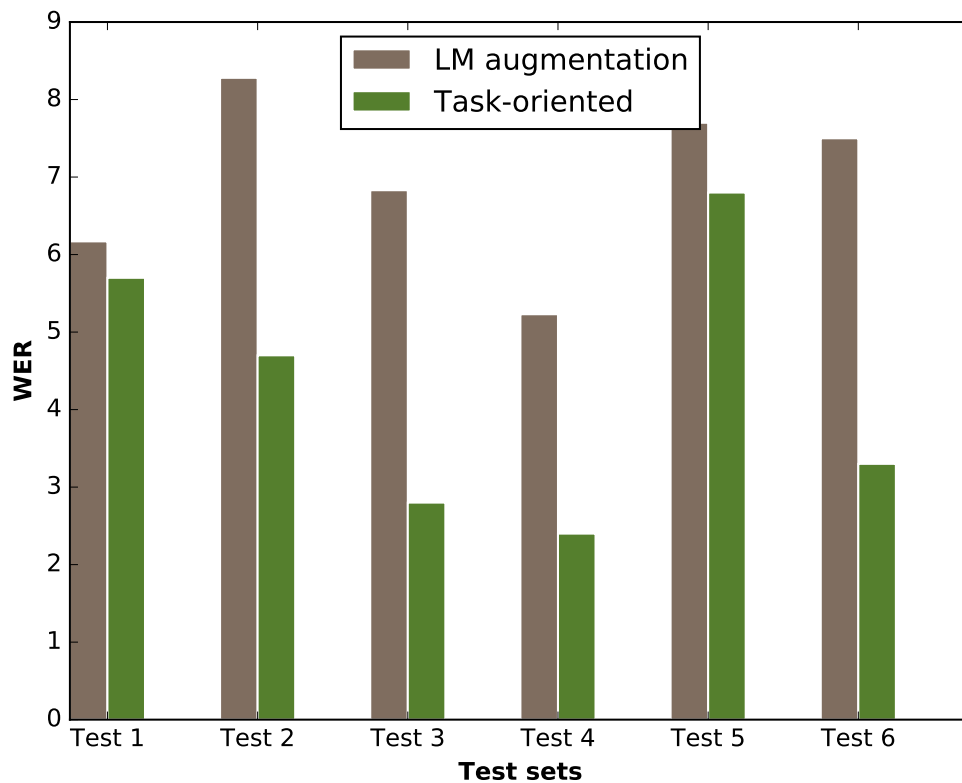


Figure 6.10 Comparison of WER improvements between the LM augmentation approach and LM built from task-oriented lattices.

Figure 6.10 demonstrates the WER improvement between the LM built from the augmented corpus with artificial stuttering data and the LM built using task-oriented lattices. The WER of the ASR is improved across all cross-validation test sets with an average of 37.7%. One explanation might be that improvements are achieved through the correction of the pronunciation of stuttering events in the pronunciation dictionary and the constraints in the LM, which positively affect both the LM and the WER and detect more stuttering events in the CV sets.

Table 6.9 Miss rate and false positive rate comparison results between the LM augmentation approach and task-oriented lattices. 'n/a' means that this event was not present in the test set.

Measures	misses	FPR	misses	FPR	misses	FPR	misses	FPR	misses	FPR
Stuttering type	S		W		R		Ph		PW	
LM AUG Test 1	85%	1.0%	33%	0.55%	40%	0%	n/a	n/a	n/a	n/a
Task-oriented-lattices Test 1	60%	0.54%	0	0.29%	25%	0%	n/a	n/a	n/a	n/a
LM AUG Test 2	46%	2.7%	40%	0.55%	67%	0%	40%	0%	n/a	n/a
Task-oriented-lattices Test 2	38%	1.64%	27%	1.1%	33.3%	0%	25%	0%	n/a	n/a
LM AUG Test 3	64%	1.0%	33%	0.27%	20%	0%	20%	0%	n/a	n/a
Task-oriented-lattices Test 3	54%	0.54%	33%	0.27%	20%	0%	16%	0%	n/a	n/a
LM AUG Test 4	60%	0.55%	43%	0%	0%	0%	50%	0%	n/a	n/a
Task-oriented-lattices Test 4	40%	0.27%	43%	0.27%	0%	0%	50%	0%	n/a	n/a
LM AUG Test 5	40%	2%	60%	0.82%	0%	0%	33%	0%	100%	0%
Task-oriented-lattices Test 5	25%	1.4%	20%	0.54%	0%	0%	33%	0%	50%	0%
LM AUG Test 6	75%	1.3%	40%	1.1%	40%	0%	50%	0%	n/a	n/a
Task-oriented-lattices Test 6	63%	0.27%	25%	0.82%	40%	0%	50%	0%	n/a	n/a

The results presented in Table 6.9 compare the ability of an ASR with a statistical augmented LM that is trained on stuttering data versus an ASR using task-specific lattices. As expected, a general improvement is seen after applying task-oriented lattice decoding on all classes of stuttering events. There is an average improvement of 24.3% relative to the LM augmentation approach in sound repetition. The average missing rate of word repetition in the LM augmentation approach reaches 41.5% , while the rate for task-oriented lattices is 24.6%. Therefore, the missing rate improvement is 40.7%. A phrase repetition results in a slight enhancement of 10.8%, while revisions improve by 29%. The part-word repetition only occurs in *Test5* with a 50% missing rate improvement.

There are no changes in recognising phrase repetitions in *Test4*, *Test5* or *Test6* and no changes in recognising revisions in *Test3* or *Test6*. This missing might be due to recording

quality. Both approaches do not produce false alarms in these two classes of stuttering. The LM augmentation approach is based on a statistical-trigram LM, which requires more phrase and revision examples in training sets to produce more false alarms in these two classes. In the task-oriented lattices approach, the weights assigned to the back-arcs are lower than the weights assigned to the self-loop arcs.

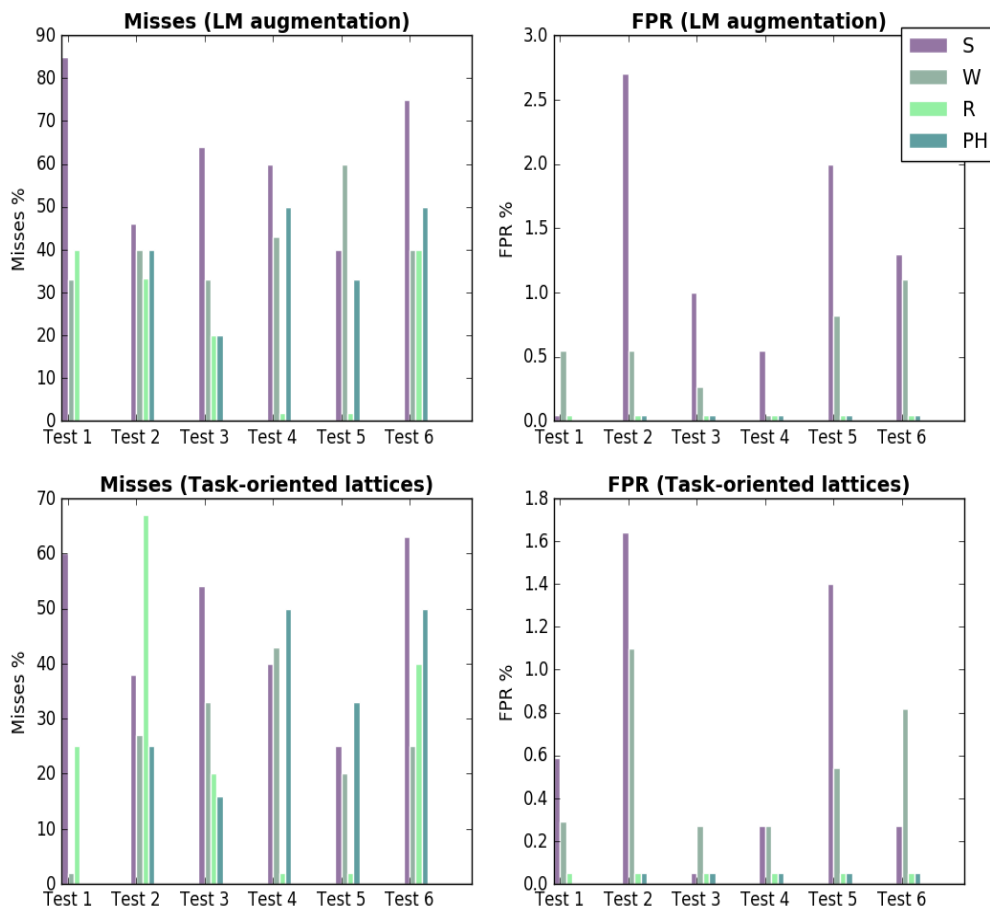


Figure 6.11 Comparison of misses and FPR improvement between the LM augmentation approach and LM built from task-oriented lattices.

Table 6.10 Average miss rate and false alarm results between the LM augmentation approach and task-oriented lattices.

Test folds	LM augmentation		Task-oriented lattices	
	misses	FPR	misses	FPR
Test 1	52.6	0.10	28.3	0.29
Test 2	39.83	0.81	39.25	0.68
Test 3	34.25	0.31	30.75	0.07
Test 4	38.25	0.14	33.25	0.14
Test 5	46.6	0.56	25.6	0.38
Test 6	51.25	0.60	44.5	0.27
Average	43.79	0.42	33.60	0.32

There is about a 23.27% improvement of the average miss rate after applying task-oriented lattices. These results are tested statistically using the t-test. We find that the  $p$ -value equals 0.0466, and this difference is considered statistically significant. The system preserves an average of 66.4% of the stuttering events after applying the task-oriented lattices approach, while the preserved stuttering events are only 56.21% after applying the LM augmentation. By contrast, the average false alarm rate decreases to 0.32%. This might be due to the constraints applied in each task-specific lattice.

## 6.6 Summary

The aim of this thesis is to produce an automatic tool that assists clinicians by providing them with a full, verbatim transcription that includes stuttering events and a count of each stuttering disfluency to provide a more accurate evaluation of the system. The verbatim transcription can help clinicians speed up the assessment process and create a dataset for further investigations.

This objective is addressed by the proposed two approaches. The first approach relies on building an ASR system with an augmented LM to recognise stuttering in recorded audio files. The second approach uses a task-specific lattice re-scoring. The results show that the task-oriented lattices out-perform the augmented LM approach by 23.27% relatively.



# Chapter 7

## Prolongation Detection System

The content of this chapter is based on conference paper previously published in (Alharbi et al., 2018).

### 7.1 Introduction

The previous chapter described two ASR approaches for the detection of most stuttering events, including sound/word/part-word/phrase repetitions and revisions. The experiments show that the ASR system using a task-oriented lattice outperforms the ASR system using LM augmentation. While ASR is highly effective in identifying segments classified by frequency-based features, it is less successful in identifying segments classified by time-based features such as prolongation events. Prolongation is the uncontrolled extension of vocalized sounds (such as rrrrunning) or of non-vocalized sounds (ssssteven).

In the current literature on prolongation classification using different methodologies, the approach by Świetlicka et al. (2013) yields the best results. Their method is based on dividing the data into equal 4-second prolonged and fluent segments for the same utterance. However, this results in the loss of two important dimensions of information: count and time. Count information refers to the number of prolongation events occurring in the given segment, and

time information indicates when this prolongation happened. Both of these measures are important when diagnosing the severity of stuttering.

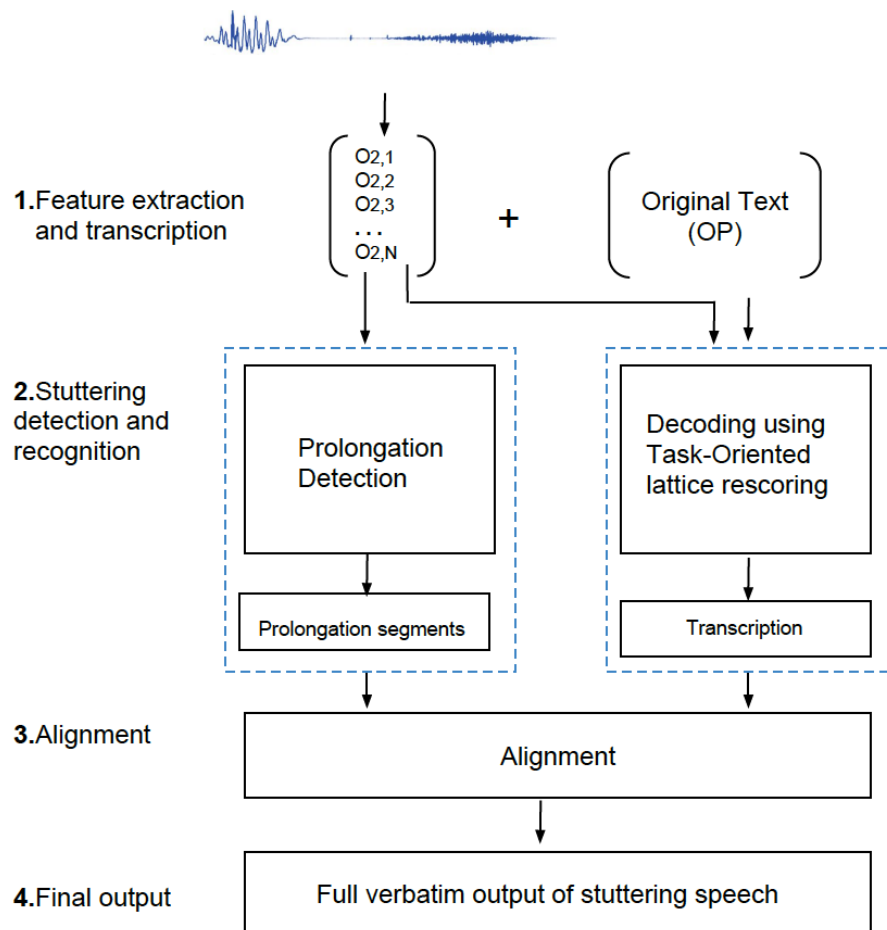


Figure 7.1 Main stages of the proposed integrated system.

The diagnosis system in this thesis needs to collect both when and how many incidences of each type of stuttering event occurred in each speech signal. One method to solve this problem is using a phone/ASR-based approach to detect any prolongation of the event and to produce this event in the transcription. This approach needs to modify the duration of phones' states in the ASR system to enable the ASR to detect the prolonged phone. Currently, the ASR only detects the normal phone duration. There is a need to re-train the ASR with a new phone duration model to be able to remain in the HMM state if the phone is prolonged.



However, applying the phone/ASR-based approach to our task-oriented lattice ASR system might affect the detection of other types of stuttering events. Because the final goal of this project is to help the therapist by providing him/her with a final stuttering severity speech sample, building an independent prolongation detection system without affecting the phone model of the task-oriented lattice ASR system is preferable.

The present work proposes a refined workflow to preserve time and count information of prolongation event. As mentioned before, to address this problem, a separate system is built to detect rather than classify prolongation segments and align these with the transcription output from ASR. Figure 7.1 shows the proposed integrated system. Speech signals are initially parameterised to delta-delta-mel-frequency-cepstral-coefficient (DDMFCC) feature sets. DDMFCC frames are subsequently analysed in parallel by a feature-based ASR trained on stuttering speech and by an autocorrelation-based prolongation detector. The ASR produces a transcription containing most types of stuttering event, such as sound, part-word, word or phrase repetitions and revisions, but not prolongations. The prolongation detector acts in parallel to detect prolongation events and serves as a correction layer for the ASR. Thereafter, all detected prolonged segments are aligned with the ASR output to produce a detailed verbatim transcript containing stuttering events.

This chapter demonstrates and compares two proposed methods for detecting prolongation events directly from a speech signal. The first of these is a supervised approach and involves two supervised machine learning techniques. The second method uses an unsupervised approach to detect prolongation segments without any need to train a classifier.

The structure of this chapter is as follows. Section 7.2 discusses the two main approaches, the first of which is based on supervised machine learning approach while the second is unsupervised. Then, experiments related to both approaches are explained in Section 7.3. Finally, a summary is presented in Section 7.4.

## 7.2 Approaches for prolongation event detection

This section describes two approaches to direct detection of prolongation in a speech signal: the supervised approach that involves two machine learning techniques and the unsupervised method. Experimental results for both methods are discussed later in Section 7.3.

Both the supervised and unsupervised approaches use Mel-frequency cepstral coefficient (MFCC) features which are considered low-level as well as the delta variations of these features. To extract the features, the speech signal is first decomposed into small frames using a 25ms analysis window with 10ms overlap. For each window, 12-dimension MFCC features are then extracted. Audio segments of different sizes would result in different numbers of low-level features. The feature extraction process is illustrated in Figure 7.2.

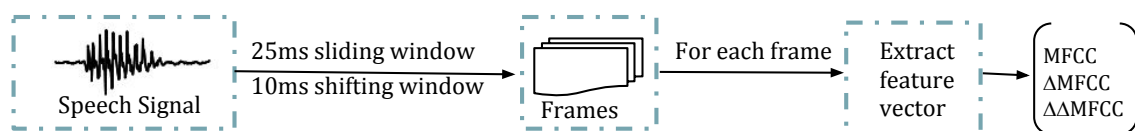


Figure 7.2 Extraction of low-level audio features.

### 7.2.1 Supervised approach

The two next subsections detail the different methods for detecting prolongation events. The first method involves training different classifiers such as k-nearest neighbors (KNN) and support vector machine (SVM) using available limited and unbalanced data. The second method represents speech signals by means of a fixed-length supervector with two-level filtration.

#### Machine learning classifiers trained on unbalanced data

As discussed in Section 5.1, there is clearly a difficulty in detecting stuttering events that include repetitions at different time scales (such as sound, part-word, word and phrase

repetitions, and time-based events such as prolongation and blocking) directly from speech signals. Based on the current limited data, it is difficult to generalize learning across stuttering events that are phonetically distinct. When counting a stuttering event, prolongation is just one event for any prolonged phone. However, as the utterance may involve different prolonged phones such as mmm, ffff, ssss and aaa, there is a need for more training data in order to be able to classify all of these as a single event (prolongation).

A further issue relates to data balance. The current limited data are also unbalanced; stuttering event classes are always fewer than fluent words, which affects the classifier training process. For that reason, we conducted a pilot study on different machine learning classifiers to test their ability to detect prolongation events directly from a speech signal, beginning with a prolongation event to test the classifiers' ability in relation to a binary classification problem (prolongation/non-prolongation) rather than a multi-classification task.

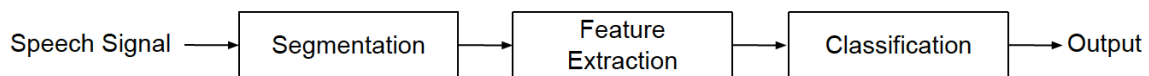


Figure 7.3 *Classification process.*

Figure 7.3 shows the three main stages of the pilot study. In the data preparation stage, prolongation and non-prolongation events are segmented manually. The second stage (feature extraction) converts speech signals to frames, providing a feature set. Finally, the classification stage applies different machine learning approaches. Section 7.3.2 describes and discusses the results using this approach.

### **GMM-Supervector approach**

The Gaussian mixture model (GMM)-supervector approach has been used for text-independent speaker recognition tasks (Kinnunen and Li, 2010; Campbell et al., 2006). The fundamental concept of this method is to re-train a universal background model (UBM) through maximum a posteriori probability (MAP) adaption and apply the adapted means of the mixed

components to construct supervectors for speaker recognition (Campbell et al., 2006). This approach could be applied to stuttering problem contexts to help the classification of a given speech utterance as either a prolongation or non-prolongation segment. The density function of a GMM is defined as

$$g(x) = \sum_{i=1}^N W_i N(x; m_i, \Sigma_i) \quad (7.1)$$

where  $W_i$  is the mixture weights,  $N(,;)$  is the Gaussian density function, and  $m_i$  is the mean and  $\Sigma_i$  the covariance matrix of the  $i_{th}$  Gaussian component, respectively. A diagonal covariance has been assumed.

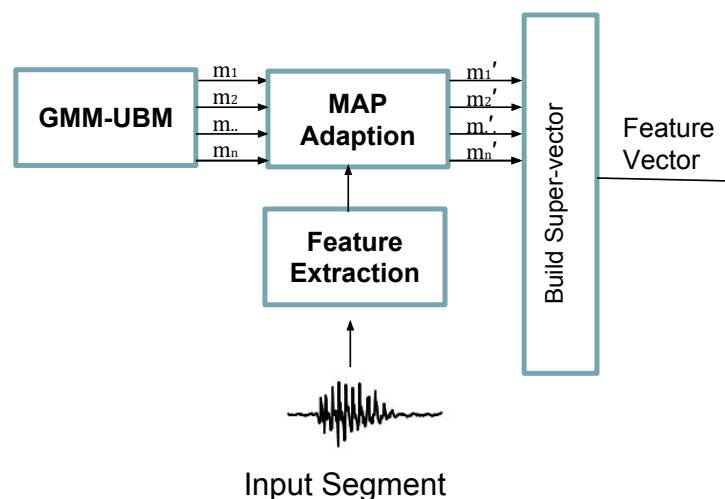


Figure 7.4 GMM supervector concept

As shown in Figure 7.4, given a prolongation utterance and using MAP adaptation (Reynolds et al., 2000) of the means, GMM training of the UBM is performed. Subsequently, the GMM supervector is derived from the adapted model. The main steps for constructing the GMM supervector in this project are as follows:

1. All recordings of children's speech are obtained from Release2 UCLASS (Howell et al., 2009) to train the UBM. To reduce the problem of unbalanced data, the training set

includes four minutes of prolongation segments and eight minutes of non-prolongation segments. Details of data preparation are provided in Section 7.3.2.

2. The UBM is calculated by GMM estimate using the general database. The UBM works as a primary and background model which is independent of the class (prolonged, or not) of the given utterance. It is applied to produce a common representation for all potential utterances. Primarily, this should be a general set of speech samples including both classes;
3. The pre-trained UBM is adapted through the MAP algorithm for each utterance to express the information provided by the current utterance;
4. The GMM supervector is created by concatenating the mean of each Gaussian component of the adapted model which take the form of

$$\mathbf{m} = \begin{bmatrix} m1 \\ m2 \\ \cdot \\ mN \end{bmatrix}$$

These GMM supervectors are then used as the input vectors to the KNN classifier.

In the recognition phase when using the GMM supervector approach, the similarity between the target ( $Y_{target}$ ) and UBM ( $Y_{UBM}$ ) models can be measured by means of the log-likelihood ratio (LLR) (Kinnunen and Li, 2010), defined as:

$$LLR(X, Y_{test}, Y_{UBM}) = \frac{1}{T} \sum_{t=1}^T \{ \log p(x_t | Y_{target}) - \log p(x_t | Y_{UBM}) \} \quad (7.2)$$

where  $X = x_1, \dots, x_T$  are the feature observations extracted from segments of the test sample T and  $p(x_t|Y)$  is the GMM density of observation  $x_t$ .

The experiments in this work demonstrate that the GMM supervector approach improves the detection of prolongation events over other classical classification methods. The results will be represented in Section 7.3, both with and without the effect of different filtration processes that were applied at a frame level

As an extension to this approach, joint-factor analysis and i-vectors can be applied which have proved to be successful in speaker verification and language identification (Dehak, Torres-Carrasquillo, Reynolds and Dehak, 2011; Dehak et al., 2009; Martinez et al., 2011). I-vector approach is a strategy for low-dimensional speaker and channel-dependent space which are represented using simple factor analysis (Kenny et al., 2005; Dehak, Kenny, Dehak, Dumouchel and Ouellet, 2011). More work could be done in the future by applying the I-vector approach on the same stuttering classification problem.

### **7.2.2 Unsupervised approach**

As an alternative to the above, we developed a novel autocorrelation detector designed to be used in parallel with the ASR system. In order to detect prolongation events, an autocorrelation function (ACF) is applied to measure the similarity between successive speech frames and proposed prolongation events as a correction to the ASR word lattice. Since stutterers usually have a lower speaking rate than normal speakers, the threshold for detecting prolongation must account for natural variations in fluency and the speaking rate on different occasions (Andrade et al., 2003). We adopted the unsupervised approach of (Esmaili et al., 2017; Suszyński et al., 2015), using two thresholds to decide whether two successive frames are similar and whether the duration of similar frames is sufficient to count as a prolongation. A threshold based on the speaking rate of the tested sample is used as one of the thresholds. However, we enhanced this method by reducing the incidence of false alarms and provide a fully automatic thresholds. Our prolongation detector uses a prior filter to mark frames judged to be silence, and removes them. The autocorrelation is then

applied to measure the similarity between speech frames as a function of the lag in order to detect prolongations. Moreover, the initial method requires a manual adjustment of speaking rates, for if the system does not accept the speaking rate threshold, there is a need to adjust the threshold manually. To solve this problem, a maximum tolerance for the speaking rate threshold is applied. According to Zebrowski (1994), prolongation ranges in length from 400 msec to 1063 msec. This means that the maximum tolerance for the speaking rate threshold is allowing a maximum of 1063 msec. Remaining similar frames are then dealt with as new candidate prolongation segments. This step reduces false alarms in cases of children who have a very low speaking rate, and it also automates the previous manual system.

### **Autocorrelation function (ACF)**

Autocorrelation function (ACF) is applied to handle prolongation events. This measures the similarity between successive speech frames and proposed prolongation events. We established empirically that the best threshold value is 0.9 when deciding whether two successive frames are similar. According to Box (1994), the autocorrelation for lag  $k$  is:

$$r_k = \frac{c_k}{c_0} \quad (7.3)$$

where  $c_0$  is the sample variance of the time series and  $c_k$  is calculated as:

$$c_k = \frac{1}{T} \sum_{t=1}^{T-k} (y_t - \bar{y})(y_{t+k} - \bar{y}) \quad (7.4)$$

The autocorrelation function measures the correlation between  $y_t$  and  $y_{t+k}$ , where the  $y_t$  and  $y_{t+k}$  are the speech signal frames, and  $k$  is the time shift, where  $k = \{0, 1, 2, 3, \dots, K\}$ .

The length of candidate segments is then normalised by a second threshold determined by a speaking rate for accepting candidates as prolongation events, helping to reduce the number of false alarms. The speaking rate detector is described in the next section.

### Speaking rate detector

As stutterers usually have a lower speaking rate than normal speakers (Andrade et al., 2003), the thresholds for determining the duration of prolongations must account for natural variations in fluency and speaking rate on different occasions.

There are different ways of estimating speaking rate from a speech signal; one of these is based on counting frequency of peak energy (de Jong et al., 2007). Others include use of a vowel classifier (Yuan and Liberman, 2010) or counting syllables by computing signal energy and zero-crossing rate (Pfau and Ruske, 1998). In all the proposed methods, speaking rate is commonly computed by counting syllable kernels segments in the signals.

For the purposes of this thesis, the speaking rate detector uses smoothed short-term energy and zero-crossing rates to detect a syllable kernel when the energy reaches a maximum in the absence of a peak in the zero-crossing rate. This provides sufficiently accurate estimates of syllables and informs the speaking rate (Pfau and Ruske, 1998).

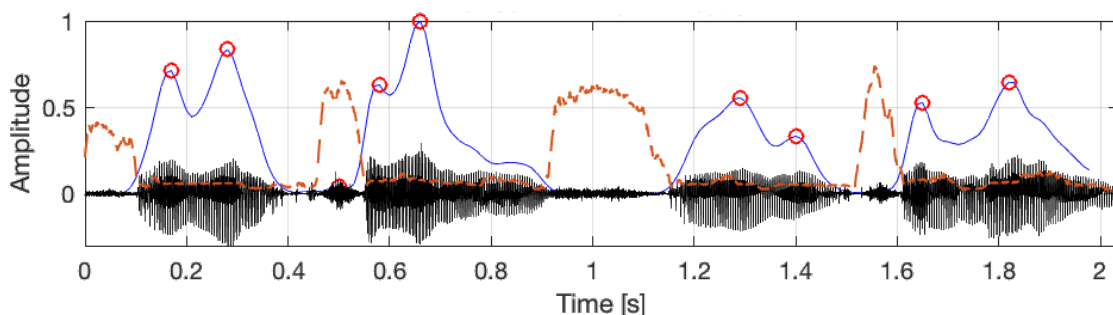


Figure 7.5 An example of syllable counting method. Solid line, and dashed line are energy signal, and zero-crossing rate, respectively. Circle marks are considered in the syllable counting process.



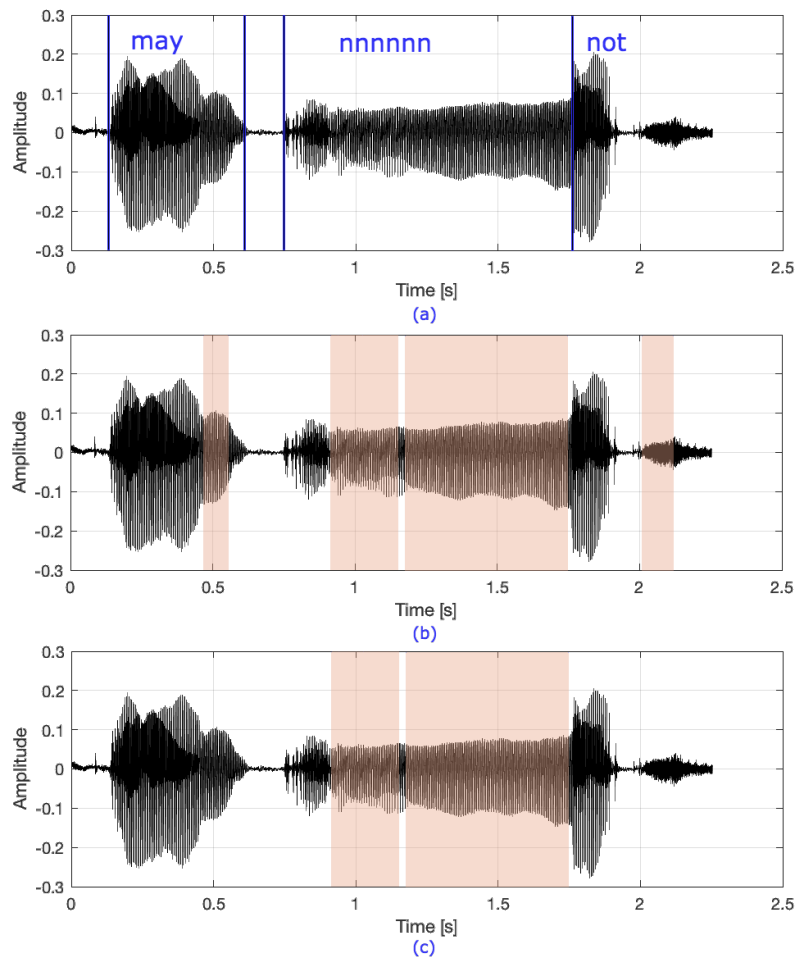


Figure 7.6 Prolongation detection. (a) Speech sample ‘may not’ with prolongation in letter ‘n’ in word ‘not’ (b) highly similar segments detected using ACF, and (c) detected prolonged segment which is longer than threshold determined by speaking rate detector.

Figure 7.6 (a) shows a prolongation segment ‘may nnnnn not’. After applying the ACF filter (Figure 7.6 (b)), all highly similar segments are selected as candidate prolongation events. Figure 7.6 (c) shows the remaining segments after applying the second filter (speaking rate). This compares the duration of each segment with the threshold determined by the speaking rate detector. If the selected segment is longer than the dynamic threshold, this segment is counted as a prolongation.

The results of unsupervised approach are below in Section 7.3.

## 7.3 Experiments

This section describes experiments involving the different methods proposed in Section 7.2 for direct detection of prolongation from a speech signal. The first subsection discusses three supervised machine learning approaches, and the second subsection presents results for the unsupervised approach.

### 7.3.1 Metrics

The conventional metrics used in the following experiments are: precision  $Prec$ , recall  $Rec$ , F1 score and accuracy  $Acc$  are used to evaluate the performance of the classifiers. The definitions of these metrics are given below.

$$Prec = \frac{TP}{TP + FP}, \quad Rec = \frac{TP}{TP + FN}$$

$$F1 = 2 \frac{Precision * Recall}{Precision + Recall}, \quad Acc = \frac{TP + TN}{TP + TN + FP + FN}.$$

$TP$ ,  $FP$  and  $FN$  refer to true positive, false positive, and false negative counts, in that order.

### 7.3.2 Supervised approach

#### Machine-learning classifiers trained on unbalanced data

In this experiment, a pilot study was conducted to test the ability of different machine learning classifiers to detect prolongation events directly from a speech signal, using 48 speech recordings from UCLASS corpus/reading, Release Two. We identified prolongation and fluent segments manually by listening to the speech recordings, and 80% of the segments

were used as a training set while 20% were used as a test set. The segmenting methodology was reviewed by UK-registered speech language therapist.

In the second (feature extraction) stage, delta-delta mel frequency cepstral coefficients (DDMFCC) were applied. Finally, in the classification stage, different machine learning approaches were applied. The author applied SVM and KNN to classify fluent and prolonged speech segments in the speech recordings test. After segmenting data (in the data preparation stage), the main problem is unbalanced data; while there were only 120 prolongation segments (which represents 0.7% of complete data), there were 16322 fluent segments.

Table 7.1 *Results of applied KNN and SVM classifiers to detect prolongation events directly from speech signal.*

<b>Classifier</b>	<b>Data</b>	<b>Features</b>	<b>Precision</b>	<b>Recall</b>	<b>F1</b>	<b>Accuracy</b>
KNN	48 UCLASS	DDMFCC	5%	40%	9%	85%
SVM	48 UCLASS	DDMFCC	20%	14%	17%	97%

Table 7.1 summarises the results of applying KNN and SVM classifiers to detect prolongation events directly from a speech signal. The results indicate poor precision, recall and f-scores. However, overall accuracy is high for both classifiers at 85% and 97% for KNN and SVM, respectively, because their training and test sets had so few prolongation segments and most segments were classified as fluent segments. Therefore, while the accuracy appears to be high, this does not demonstrate its effectiveness on detecting prolongation. Instead, focus should be placed on the precision and recall scores.

Table 7.2 *Comparison of achieved results with the results in the literature. n/a means not mentioned.*

Author	Classifier	Data	Features	Precision	Recall	F1	Accuracy
Pilot study	KNN	48 UCLASS	DDMFCC	5%	40%	9%	85%
Pilot study	SVM	48 UCLASS	DDMFCC	20%	14%	17%	97%
(Mahesha and Vinod, 2016)	GMM	50 UCLASS	DDMFCC	n/a	n/a	n/a	95.70%
(Ai et al., 2012)	KNN	39 UCLASS	LPCC	n/a	n/a	n/a	94.51%
(Ai et al., 2012)	KNN	39 UCLASS	MFCC	n/a	n/a	n/a	92.55%
(Chee et al., 2009)	KNN	10 UCLASS	MFCC	n/a	n/a	n/a	87%
(Chee et al., 2009)	LDA	10 UCLASS	MFCC	n/a	n/a	n/a	85%

Table 7.2 confirms that the high accuracy results agree with results reported in previous work that attempts to classify prolongation events directly from a signal based on similar (UCLASS) unbalanced data. However, we have already argued that the accuracy score is not especially relevant because the majority of the speech classified was fluent in the test set. Studies by (Mahesha and Vinod, 2016; Chee et al., 2009) also do not include any further statistical data, thereby making it difficult to carry out a full comparison without their precision and recall scores.

### **GMM-Supervector approach**

This section presents the results from the GMM-supervector approach because previous similar studies sought to minimize the unbalanced data problem. The prolongation detector system forms part of the proposed integrated system illustrated in Figure 7.1, based on a standard reading task used by clinicians to diagnose stuttering in children. All recordings of children's read speech were obtained from UCLASS Release Two (Howell et al., 2009).

In the preparation stage, the author used Audacity software to assign all data to prolongation and non-prolongation segments, and each segment was cut to 400 msec. According to Zebrowski (1994), the minimum duration of sound prolongation is 400 msec. It follows

that cut each segment to 400 msec will cover the minimum prolonged segment. Overlapping segments that contain a mix of prolongation and non-prolongation frames are labelled as prolongation segments if the segment frames are more than 60% similar (prolonged).

To reduce the problem of unbalanced data, the training set included a total of four minutes of prolongation segments and eight minutes of non-prolongation segments, cut out in both cases to 400 msec. For testing, we used the set of speech recordings (with ‘Arthur the rat’ passages) similar to the Section 7.3.3 (Unsupervised approach) which ‘Arthur the rat’ passages have been used. All test speech recordings were cut out to 400 msec and labelled as prolongation or non-prolongation as a ground-truth for evaluating the performance of the classifier. Overlapping segments were treated as prolongations if more than 60% of frames were similar.

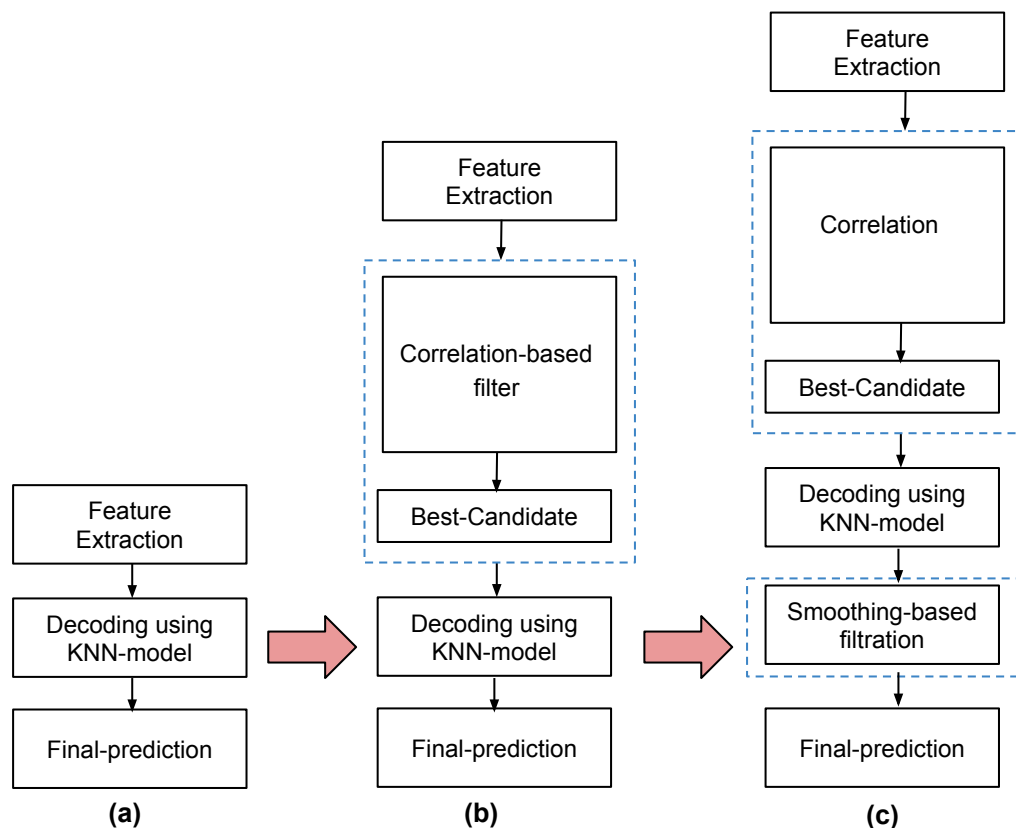


Figure 7.7 (a) Decoding stage without filtration process. (b) Decoding stage with correlation filter process. (c) Decoding stage with correlation and smoothing-based filtration process.

In the decoding stage, we examined the results in three different cases as shown in Figure 7.7. The first case includes the results for the GMM-supervector approach without any effect (without applying filters). In the decoding stage, two different filters were applied for enhancement. The first was a correlation-based filter that measured the similarity between frames and provided the best candidate segments for the decoding stage using the KNN classifier. After feature extraction, the correlation of each segment was calculated using ACF. Correlated segments are the best candidates for decoding using the KNN classifier. The KNN classifier will classify this segment if it is a prolongation or it is a correlated segment came from noise or silence. If the segment was not correlated, it was immediately classified as non-prolongation and did not need to be decoded using the KNN classifier.

The second filter is a type of smoothing filter. As the KNN decodes at frame level, the classifier labels each frame as either prolongation (1) or non-prolongation (0). The smoothing filter then looks at the two previous frames and the two subsequent frames for each frame in the segment and labels the frame accordingly. For example, if the current frame is labelled 0 while the previous frame and the next two are labelled 1, the smoothing filter will label this frame as 1.

Table 7.3 *Results after applying GMM-supervector approach in three different cases during decoding stage. The first case demonstrate the results without applying any filters. The second results after applying a correlation-based filter. The third results after applying a correlation and smoothing filters.*

<b>Filter</b>	<b>Precision</b>	<b>Recall</b>	<b>F1</b>
No filters	12%	75%	21%
Correlation-based-filter	15%	68%	25%
Correlation/smoothing-based-filter	88%	58%	70%

Table 7.3 presents the results after applying the GMM-supervector approach. The first experiment applied KNN without filters. The results indicate high recall but low precision,

with an f-score of 21%. After applying a correlation-based filter, recall fell to 67%; precision increased to 15%, and the total f-score increased to 25%. Overall results improved after applying the second filter, and the f-score reached 70%.

The GMM-supervector based achieved a great improvement of over a ML approach presented in the previous section in precision and recall. As mentioned before, for a diagnostic system that determines whether patients should receive treatment, recall is more important than precision, which would fail to diagnose patients who genuinely required treatment (Lever et al., 2016; Powers, 2011).

In this section, different supervised approaches have been conducted, which are largely based on classification rather than detection. In the GMM-supervector approach, we tried to build a detector by having a sliding window through the test set and apply a classifiers to perform a binary classification of a 400 msec frame of speech to classify whether it was or was not a prolongation segment. Then, merging any segments that where identified as sequential piece of prolongation. However, they are still not achieving good results with a binary classification task as the higher f-score we achieved after applying a GMM-supervector approach, however, with recall of 58% which is inadequate for the diagnosis framework. This is also an indication of deploying these approaches to perform a detection task over different time scales in a continues speech to detect a different types of repetition, its most likely to be even worse.

Therefore, we decided to switch to an unsupervised approach to detection of prolongation events without any need for training classifiers with limited data.

### 7.3.3 Unsupervised approach

This section reports results for an ACF with speaking rate detector. As mentioned earlier, the system is based on a standard reading task used by clinicians to diagnose stuttering in children, using recordings of children's read speech obtained from UCLASS Release Two (Howell et al., 2009). All recordings of 'Arthur the rat' used for testing ASR performance were analysed and applied as an input to test the performance of the prolongation detector system and to detect prolongation segments. Using Audacity software, all recordings were assigned by the author to fluent or prolongation classes. In total, 35 speech samples of sound prolongations and 9,400 speech samples of fluent words were obtained. The labelling methodology was reviewed by two UK-registered speech language therapist.

When evaluated using the test set, the prolongation detector successfully identified most prolongation events and achieved 92% recall. The results clearly indicate that detection of similarly correlated successive frames results in the effective identification of prolongation during continuous speech. However, artefact noise (such as background noises and heavy breathing during recording) was still erroneously identified as prolongation. Using this approach with our existing silence remover and after restricting speaking rate threshold tolerances to allow a maximum of 1063 msec of prolongation achieved 60% precision, reducing false alarms when using the proposed method for children with a very low speaking rate.

However, the current method is not useful for detecting all prolongation segments produced by children with severe cases because their prolongations duration is sometimes more than 1063 msec (Zebrowski, 1994). Fortunately, only 3 out of 25 'Arthur the rat' recordings were diagnosed as severe cases by a UK-registered speech language therapist.



## 7.4 Summary

This chapter discussed two main approaches to building a separate prolongation detector system. The first of these was a supervised approach that proposed different solutions to the unbalanced data problem and improved precision (88%) and recall (58%) by applying a UBM-based approach. However, the supervised approach appeared inadequate, as 58% recall is insufficient for diagnostic purposes. The unsupervised approach is based on ACF, which measures the similarity between successive speech frames and applies several thresholds to filter the duration of each detected segment. This approach improved recall to 92%, but precision deteriorated to 60%. In the future, detected false alarms could be further reduced by applying a better silence remover.

Chapter eight includes an analysis of the transcript produced by the proposed integrated system and an evaluation of the ability of machine learning approaches to detect stuttering events from that transcript.



# Chapter 8

## Detecting and Classifying Stuttering

### Events in Transcriptions

The content of this chapter is based on conference paper previously published in (Alharbi, Hasan, Simons, Brumfitt and Green, 2017).

#### 8.1 Introduction

Literal transcriptions of patients' speech can facilitate the detection of different types of stuttering events. Archived transcriptions are also useful for further investigative research into the condition. As mentioned previously, recording and manually transcribing stuttering speech is tedious; it requires significant time and effort due to the need to chronicle each spoken word. Therefore, the use of automatic speech recognition (ASR) could expedite the assessment of children's speech and make the diagnostic process more efficient. Also, it could be easier to archive data for further evaluation. This has motivated the need for an ASR system that produces word-level transcriptions and a classifier that detects and classifies stuttering events in the acquired transcriptions.

In previous chapters, we proposed an ASR system that can produce a nearly verbatim transcription, including different orthographic forms for phonetic occurrences of stuttering. This allows stuttering events to be counted by a classifier that is trained to detect them.

Previous studies have built in automatic disfluency detection to remove disfluencies from transcriptions (Honal and Schultz, 2003; Snover et al., 2004) because it is important for making the ASR output more readable and for aiding downstream language processing modules. Our goal differs in that we do not seek to remove stuttering events but to provide a final count of each stuttering event that is recognised by our proposed framework.

The work in this chapter investigates the detection of stuttering events from transcription. One suggested approach is a simple traditional rule-based (RB) algorithm. Traditional RB algorithms are powerful in transferring the experiences of domain experts to make automated decisions. This approach could be applied for event detection tasks. For offline applications where time and effort are not concerns, it can work with high accuracy for limited target data. However, this approach depends on the expert's knowledge (Liu et al., 2016), which means it only works if all situations of stuttering events are considered. This condition cannot be satisfied in practice due to the continuous variability in data volume and complexity. Moreover, this knowledge based approach is deterministic as it uses rules like "If word  $W$  is preceded by word  $Z$ , within  $C$  number of words, trigger the event  $Y$ ", and if such scenarios are missed, false absolute decisions will be made, which might not happen under a probabilistic training regime.

Alternative probabilistic approaches are therefore required to learn the rules from the structure embedded in the data (i.e the stuttering pattern encapsulated in the stuttering sentences). Machine learning classifiers such as conditional random fields (CRF) and bi-directional long-short-term memory (BLSTM) can actually help build data driven rules, and furthermore, as we find more data, these classifiers can be easily and frequently retrained.

This chapter evaluates the ability of two machine learning approaches CRF and BLSTM to detect stuttering events from both human and ASR transcripts of children's read speech and compares the performance between them. This comparison is conducted using lexical, and contextual features. In addition, this chapter studies the effect of adding more data to improve the classifiers' performances. This study also investigates the effect of augmenting the available training data with artificially generated training data to improve the classifiers' performances. Finally, this chapter describes a method for studying the effect of ASR errors on classifiers' performances.

The structure of this chapter is as follows. Section 8.2 presents a review of the related literature; Section 8.3 define the task of this chapter; Section 8.4 describes the CRF and BLSTM classification approaches; Section 8.5 presents the feature engineering and extraction processes; Section 8.6 presents the experimental setup and results; Section 8.6.1 describes the guidelines and methodology used to produce the stuttering data transcriptions and annotations; Section 8.7 presents the conclusion and recommendations for future research.

## **8.2 Previous work**

Several disfluency detection systems have been introduced in the past (Honal and Schultz, 2003; Snover et al., 2004; Liu et al., 2005; Honal and Schultz, 2005; Maskey et al., 2006). and these systems are powerful for creating more readable output from speech recognition as well as assisting downstream modules of language processing. However, our aim is to detect and classify stuttering events to facilitate the counting of each event.

During the assessment phase in a therapy session, clinicians need to carefully measure stuttering events to determine the severity level of stuttering. This measurement is usually done by counting the number of stuttering events in the child's speech during the real-time session. Notably, proper measurement is extremely dependent on the clinician's experience (Blood et al., 2010; Brundage et al., 2006; Lass et al., 1989; Brisk et al., 1997). In another

approach, the clinician transcribes a recorded session and classifies each spoken term into one of several normal, disfluent or stuttering categories (Gregory et al., 2003; Guitar, 2014; Yairi and Ambrose, 2005). Therefore, if we start from the premise that the speech has already been transcribed by the therapist, the task is then to detect and classify stuttering events within the transcriptions. Mahesha and Vinod (2015) used a lexical rule-based (RB) algorithm to detect stuttering events in orthographic transcripts from the University College London Archive of Stuttered Speech (UCLASS) (Howell et al., 2009). In particular, they used prior domain knowledge to construct expert-based sets of rules to count the number of occurrences of each stuttering event. For event detection tasks, the traditional RB algorithm is a beneficial technique for conveying the experiences of domain experts allowing them to make automated decisions. However, this approach depends on the expert's knowledge being complete, the rules fully covering every possible stuttering event, and articulation of the rules supporting diagnosis without rule-conflicts (Liu et al., 2016). Our present study demonstrates an evaluation of two machine-learning approaches, CRF and BLSTM for detecting stuttering events both in human and ASR transcripts of children's read speech by building data driven rules rather than the rules obtained from expert's knowledge.

Initially, we train probabilistic models to detect stuttering events in human transcriptions of stuttering speech, on the basis that the human transcription is the gold-standard and will give the best training data. Our eventual goal is to detect stuttering in ASR transcripts, which may include ASR errors. Some studies have investigated how ASR errors affect classification results (Liu et al., 2005). It is well known in the literature that, for different classification tasks, the performance of classifiers decreases in the presence of ASR errors. For example, errors in the ASR output caused a degradation in the performance of the Hidden Markov Model (HMM) and the Maximum Entropy Model (MaxEnt) by 50.6% and 52%, respectively in a disfluency detection task, using the broadcast news (BN) speech corpus (Liu et al., 2005).

### 8.3 Task definition

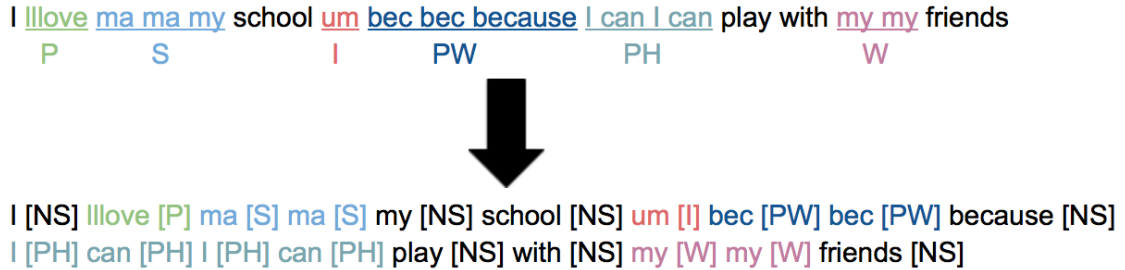


Figure 8.1 *Word level annotation example.*

Given a sentence (sequence of proper/stuttered word entities), the task is to assign a stuttering label (*I*, *W*, *PW*, *S*, *PH*, *P* or *NS*) to each entity in the sentence. Some entities have a unique label, such as the interjection words. Others could have different labels that vary with the location in the sentence. Figure 8.1 illustrates some examples of these situations. In this way, determining the correct stuttering label for each entity depends on the context of the sentence, including the labels of the neighbouring entities. Therefore, the task of detecting stuttering events in a transcription can be defined as a sequence labelling task, and approaches, such as CRF and BLSTM, can be used. The task can be formulated as follows: Given a sequence of observations/feature vectors, find an appropriate label sequence for the observations. The following section describes the CRF and BLSTM approaches used in the present study.

## 8.4 Machine learning classifiers to detect stuttering event from transcription

The motivation for applying a CRF and BLSTM for detecting and classifying stuttering events from transcriptions is the capability of these two classifiers in sequence labeling task.

Conditional random fields (CRF) is a linear statistical model which is considered as one of the most traditional high-performance sequence labeling model (Tseng et al., 2005; Ratnov and Roth, 2009; Passos et al., 2014; Luo et al., 2015). This classifier is powerful for sequence labeling task because it takes into account the correlations between neighboring labels and simultaneously predicts the best sequence of predicted labels for a given input sentence. Therefore, we model the CRF for the stuttering detection problem by designing different features to detect and classify sound, word, and phrase repetition (explained in Section 8.5) which observes the entire represented region, similar to the named entity recognition (NER) task (McCallum and Li, 2003).

Recently, non-linear models such as neural networks with word representations which use distributed input (known as word embeddings), have been largely used in problems related to natural language processing (NLP) with great success (Huang et al., 2015; Chiu and Nichols, 2015; Hu et al., 2016).

LSTM units were proposed by Hochreiter and Schmidhuber (1997), and they are a variation of the recurrent neural network (RNNs) that are able to capture long-term dependencies with the guidance of the particular structure (Bengio et al., 1994). In the stuttering labeling task, we are not only aiming to capture the past features, but there is also a need to capture the future features to detect different stuttering events such as sound, word, part-word, and phrase repetition. Therefore, we are employing the BLSTM structure used by Graves et al. (2013), which allow us to apply the forward and backward steps. This property makes use of both past features and future features.



### 8.4.1 Conditional random fields

As mentioned before, conditional random fields models are discriminative models that have been intensively used for sequence labelling and segmentation purposes (Tseng et al., 2005). This model aims to estimate and directly optimise the posterior probability of the label sequence, given a sequence of features (hence the frequently used term direct model).

Given a set of observations (sequence of words that may include some stuttered words), a CRF model predicts a sequence of labels  $y$  for these observations. Let  $X, Y$  be the observation and label sequences, respectively, and  $f(x, y)$  be the set of feature functions, the CRF model can be represented by the following equations:

$$p(y|x, \lambda) = \frac{\exp(\lambda^T f(x, y))}{Z(\lambda, x)},$$

where  $\lambda$  is the model's parameters; one weight ( $w$ ) for each feature. These weights are learned during training such that:

$$\lambda = \arg \max_{\lambda} p(Y|X, \lambda),$$

the label sequence can then be predicted from the following equation:

$$y^* = \arg \max_y p(y|x, \lambda) \arg \max_y p(y|x, w)$$

### 8.4.2 Bidirectional LSTMs

LSTM units are a variation of the RNN that overcomes the problem of vanishing gradient (Bengio et al., 1994). This property allows LSTMs to capture long-term dependencies without arithmetic problems. In particular, LSTM incorporates gated memory cells by

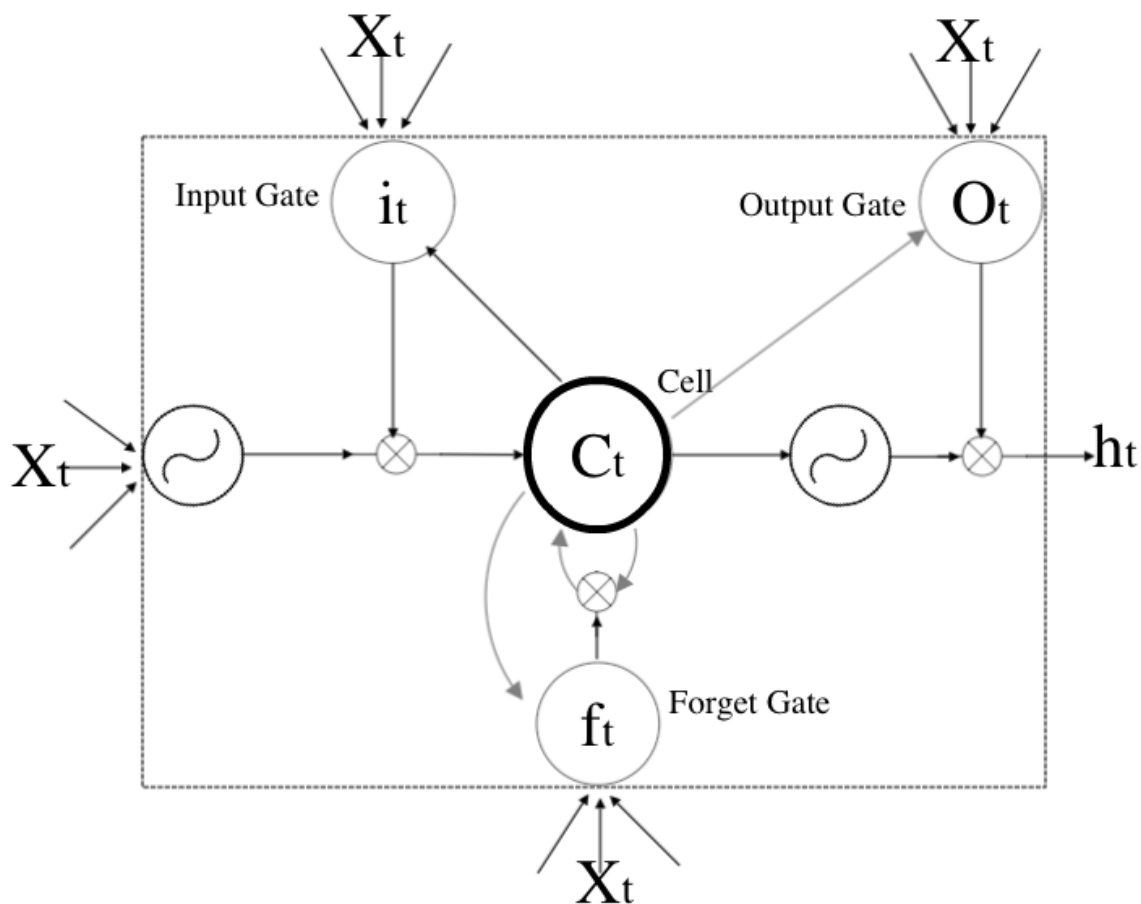


Figure 8.2 *Long Short-term Memory Cell.*

which information/signals can be read, written, deleted or stored. These operations are controlled using a sigmoid function that performs element-wise operations. The decisions in this process are based on weights that are learned during the recurrent network training. Figure 8.2 illustrates the data flows in a memory cell. The different LSTMs gates can be modelled using the following formula to update an LSTM unit at time  $t$  are (Gers et al., 2002):

$$\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i), \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f), \\
c_t &= (f_t c_t - 1) \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o), \\
h_t &= o_t \odot \tanh(c_t),
\end{aligned}$$

where  $f_t$ ,  $c_t$ ,  $i_t$ ,  $o_t$ ,  $h_t$  are the forget, cell, input, output gates and hidden state, respectively.  $X_t$  is the input vector at the  $t$ .  $W$  and  $b$  are the weights and the biases vectors of the network, respectively.  $\sigma$  is the element-wise sigmoid function and  $\odot$  is the element-wise product. To form the final BLSTM representation, the left-to-right  $\vec{h}_t$  and the right-to-left  $\overleftarrow{h}_t$  input representations are concatenated  $ht = [\vec{h}_t; \overleftarrow{h}_t]$ .

## 8.5 Features of the classifiers used to detect stuttering events

Assigning a label to a word entity is based on a set of observations, associated with this label. These observations are introduced to a classifier as a set of feature vectors. The role of the classifier is to map the set of feature vectors to a specific label, and this is done by implementing a set of steps that varies with different classifiers.

This section describes the features used by the proposed classifiers to detect the stuttering events in transcriptions. This includes uni-gram, bi-gram, tri-gram and 2-post-words for each word, as well as character- and utterance-based features for the CRF classifier and pre-trained word embeddings for the BLSTM classifier.

### 8.5.1 Word/Utterance-based features

This work introduces long-range statistics by measuring the backward distance at two levels. The first level uses a backward distance to measure how far the current word is from its

similar neighbouring words, in the word sequence. For example,

<i>sequence</i>	<i>sa</i>	<i>sa</i>	<i>sa</i>	<i>sound</i>
<i>distance</i>	0	1	2	0

This feature aids the classifier to observe the repeated patterns in the text. In the second level, we measure and compare the backward distance of each neighbouring words of two and three grams. For example, (*it is it is*) the assigned counter for the second *it* and *is* will be 1, which is an indication of a phrase repetition of two grams. For consistency, equivalent operations must be performed on the training and evaluation sets.

### 8.5.2 Character-based features

This feature were extracted in two levels. The first level uses a backward distance to measure how far the current character is from its similar neighbouring character, in the same word. For example, '*mmmay*'. This feature helps the classifier to observe the repeated characters in the same word which indicate of prolongation event. In the second level, we measure and compare the backward distance of each neighbouring characters of two and three grams for two successive words. For example, (*par particular*), the assigned counter for the characters of the second pseudo word *par* within word *particular* will be 1,1,1.

### 8.5.3 Word embedding

A meaningful word representation can be learned using neural networks from random word embedding. Word embedding is employed by converting all words into numerical vectors in a vector space by assigning all similar words (semantically) to similar vectors. A well-known algorithm, GloVe (Pennington et al., 2014), was used to perform the word embedding technique. This algorithm builds word embeddings by searching through the training data to find co-occurrences of words with the assumption that the meaning of a word usually

depends upon its context. In the present study, we used pre-trained GloVe model to generate word embeddings for each utterance. This model was trained on Common Crawl corpus of web-crawl data (having a vocabulary of 1.9 million words).

## 8.6 Experiments

The first experimental sets were designed to compare the performance of the CRF and BLSTM classifiers in relation to human transcripts. We selected varying amounts of training data from the different speaking tasks and studied how that addition was reflected in the performance of the classifiers. These investigations were conducted with and without the proposed characters/word and utterance features for the CRF classifier, presented in Section 8.5. In addition, we studied how the classifiers are affected by speech recognition errors, such as deletions, substitutions and insertions. For this, the evaluation method based on two references. The first reference is the labelled human transcript (Human\_scoring) of the test set, and the second (ASR\_scoring) is the ASR transcripts that were labelled manually using the same guidelines as those of Human\_scoring; both follow Yairi and Ambrose's (2005) annotation approach.

Because this study was conducted before the completion of the investigation of all ASR approaches that were presented in Chapters five and six, all presented experiments in this chapter were conducted on a different test set than what was applied in the previous chapters. The ASR experiments used 'Arthur the rat' passage; this chapter used an earlier set of read passages. However, for consistency and to diagnose the severity level of the stuttering of children who recorded the 'Arthur the rat' passage, we also ran additional test on the 'Arthur the rat' passage.

The first section discusses the data that are applied to train and test these experiments, followed by a discussion of the results that are found after applying CRF and BLSTM classifiers to different tasks for human transcription. We also study the effect of ASR errors

on both classifiers. Finally, the best classifier is applied to 25 recordings of the ‘Arthur the rat’ passage.

### 8.6.1 Data transcription and annotation

Table 8.1 *Types of stuttering.*

Label	Stuttering Type
I	Interjection
S	Sound repetitions
PW	Part-word repetitions
W	Word repetitions
PH	Phrase repetitions
P	Prolongation
NS	Non-stutter

The present study is based around a standard reading task that is used by therapists to diagnose stuttering in children. For training purposes, we obtained the recordings of children’s read speech from the UCLASS stuttering corpus, Release Two (Howell et al., 2009). As mentioned in Chapter four, these 48 recordings did not have any associated transcriptions. However, UCLASS Release One (Howell et al., 2009) contains another dataset of spontaneous stuttering speech, for which 31/63 recordings had transcriptions. We transcribed the remaining data following the same conventions used for this subset, and also transcribed the read speech dataset in the same way. Transcriptions were orthographic, and included conventional forms to represent stuttering disfluencies, for example: (*‘This is a a a amazing’*).

In terms of the volume of training data for this study, there were 48 recordings of read speech (Read), taken from 48 males aged between 8 and 18 years; and 63 recordings of spontaneous speech (Spon.) taken from 45 males and 18 females aged between 7 and 17 years. The manual transcriptions of this data were later used to build a language model. Since this combined dataset was still relatively small, we also examined the effect of data

augmentation, adding artificially generated transcription data (Art.) designed to increase the likelihood of stuttering events in the language model.

The present study used the SRILM toolkit (Stolcke et al., 2002b) to generate additional stuttering sentences from two inputs: a language model, trained on the UCLASS-Release1 training set, and a large word list. This word list was created by merging the UCLASS-Release1 word list with another publicly available word list *lm-csr-64k-vb-3gram* (Vertanen, 2007) which we augmented by rule with stuttering events.

The original transcription files for the recordings obtained from UCLASS (111 files from read and spontaneous tasks) with the produced artificial data were then annotated to include the stuttering type for each word using the annotation approach proposed by Yairi and Ambrose (2005). Figure 8.1 shows an example of the word-level annotation process that already explained in Chapter four, Section 4.5.

The present study considers the detection and classification of sound, part-word, word and phrase repetitions as well as interjections and prolongations. Each of the types of stuttering examined in the study are listed along with their corresponding abbreviations in Table 8.1. The transcription and annotation methodology was reviewed by two UK registered speech language therapists. Moreover, the complete transcribed text was normalised. Text normalisation is considered to be a prerequisite step for many downstream speech and language processing tasks. Text normalisation categorises text entities, such as dates, numbers, times and currency amounts, and transforms them into words.

In order to evaluate the ability of machine classifier approaches to detect and classify stuttering events from transcriptions obtained from a reading task, we partitioned the available read data into training (80%) and test (20%) sets, and we deliberately ensured that the training and test sets had relatively equal distributions of stuttering events from the start (Table 8.2). We trained initially only using the read data (*read*), then on the read and spontaneous data (*read+Spon*), then on the artificial data (*read+Spon+Art*). Statistics for the training sets that

included read/spontaneous tasks and a third training set with artificial stuttering events are also shown in Table 8.2. Table 8.3 presents the distribution of each type of stuttering event in the evaluation set.

Table 8.2 *Statistical data for the training sets.*

Task	Training Data	Words	%I	%W	%PW	%S	%PH	%P	%NS
Task <sub>1</sub>	Read	13134	0.22	1.9	0.33	1.9	1.5	1.3	93.6
Task <sub>2</sub>	Read + spon.	24137	1.8	1.9	1.0	5.3	1.2	1.4	87.4
Task <sub>3</sub>	Read + Spon. + Art.	74198	1.8	1.8	1.4	4.0	1.0	1.0	89
Average		111469	1.3	1.9	1.00	3.7	1.2	1.2	90

Table 8.3 *Statistical data for the test sets from the human transcripts.*

Set	Words	%I	%W	%PW	%S	%PH	%P	%NS
Test	3189	0.3	1.2	0.7	1.00	1.6	0.44	94.8

### 8.6.2 CRF classifier: effect of adding more data

The baseline CRF classifier was trained using the words  $n$ -grams, where  $n = 1, 2, 3$ , and two post-word features were extracted from the read speech data. There were three sets of experiments. In Task1, only the read speech data was used for training. In Task 2, this was supplemented by the spontaneous speech data; and in Task 3, was further supplemented by artificial data. The details of the artificial data are given in Section 8.6.1 and the distribution of stuttering events in all used data are presented in Table 8.2.



Table 8.4  $CRF_{ngram}$  results trained on different tasks, using  $ngram$  features only.  $CRF_{ngram}$  trained on  $Task_1$  which only using the (read) data, then on  $Task_2$  which using the (read+Spon) data, then on  $Task_3$  which using the (read+Spon+Art) data.

St-type	Task <sub>1</sub>			Task <sub>2</sub>			Task <sub>3</sub>		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
I	1.00	0.44	0.62	1.00	0.56	0.71	1.00	0.67	0.80
W	1.00	0.50	0.62	1.00	0.50	0.62	1.00	0.50	0.62
P	1.00	0.14	0.25	1.00	0.14	0.25	1.00	0.14	0.25
PH	0.52	0.27	0.36	0.65	0.25	0.37	0.56	0.23	0.33
PW	1.00	0.10	0.17	1.00	0.05	0.09	1.00	0.00	0.00
S	1.00	0.59	0.74	1.00	0.83	0.91	1.00	0.93	0.96
Average	0.92	0.34	0.46	0.94	0.39	0.49	0.76	0.41	0.49

Table 8.4 shows the results of three CRF classifiers on the detection of each type of stuttering event. These classifiers differ in the type/task and amount of data used for training, as shown in Table 8.2. One can clearly observe that, with the addition of stuttered spontaneous data, the precision results have either improved ( $PH$  by 20%, relatively) or not changed for all stuttering types. The recall results have either improved ( $I$  by 21.4%;  $S$  by 28.9%) or deteriorated ( $PH$  by 7.4%;  $PW$  by 50%), resulting in the average  $F_1$  measure for those labels. These results can be interpreted when linked to the change in the distribution of each class, after adding the spontaneous speech data. In particular, improvement in the detection of  $I$  and  $S$  is linked to the increase in their distribution ( $I$  by 87.8%;  $S$  by 64.2%), as shown in Table 8.2. However, the deterioration in detecting  $PH$  is due to this being a lower proportion of the larger training set in which its distribution was reduced by 20%.

Similarly, the  $PW$  detection was not improved. This is mainly due to the fact that words in this category have a wide range of variation because they are literally sub-units of the

words. For instance, in cases, such as ‘*pla* [PW], *play* [PW]’, which only accrues in the evaluation set, the CRF will not be able to detect it because it was not seen in the training set (i.e. out of the domain vocabularies (OOV)). The results in Table 8.4 also suggest that adding more data, without additional features, will not improve the detection of *PW* events unless the additional data contains the *PW* words that are in the evaluation set. Moreover, adding more data that affect the distribution of other classes may negatively affect the detection of these types of word-dependent classes. On average, adding spontaneous data increased the overall averages of all the measures (recall by 12.8%, precision by 2%,  $F_1$  by 6%).

The last set of results shows the outcome of adding artificial data (Table 8.4). The only class that benefits from this addition is the sound repetition class. This result is expected from the distribution reported in Table 8.2, which shows improvement only in the *S* and *PW* classes. The *PW* deterioration is due to the same reason discussed earlier, and the only solution for this type of dependency is to use additional features to help the classifier detect unseen part-words. Thus, those results motivate the introducing of context based features to reduce the dependency on the word fragments represented in n-grams and post-words features. This is addressed in the next section.

### Adding character- and utterance-based features

Table 8.5  $CRF_{aux}$  results trained on different tasks, using auxiliary features.  $CRF_{aux}$  trained on  $Task_1$  which only using the (read) data, then on  $Task_2$  which using the (read+Spon) data, then on  $Task_3$  which using the (read+Spon+Art) data.

St-type	Task <sub>1</sub>			Task <sub>2</sub>			Task <sub>3</sub>		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
I	1.00	0.78	0.88	1.00	0.78	0.88	1.00	0.78	0.88
W	0.95	1.00	0.97	0.95	0.97	0.96	1.00	0.97	0.99
P	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
PH	0.72	0.72	0.72	0.72	0.72	0.72	0.88	0.70	0.77
PW	0.63	0.57	0.60	0.53	0.71	0.63	0.82	0.94	0.87
S	0.96	0.79	0.87	1.00	0.93	0.96	1.00	1.00	1.00
Average	0.88	0.81	0.84	0.86	0.85	0.86	0.95	0.88	0.92

As shown in Table 8.5, the performance of the proposed CRF classifier with the word feature and the character/utterance-based features (described in Section 8.5) has improved over all previous stuttering types, leading to better results in the classification task. For critical classes that mainly depend on word representations, such as *PW* and *P*, adding character-based features helped detect all words with repeated characters *P*, such as *mmmay*. Moreover, this feature enhanced the *PW* classification so that it achieved a 60%  $F_1$  score on the read task, a 63%  $F_1$  score on the mixed read/spontaneous task and an 85%  $F_1$  score after adding more artificial stuttering data. Adding an utterance-based feature improved the ability of the CRF classifier to detect phrase repetitions by 36% in comparison to the baseline experiment. Adding a word-distance feature enhanced word repetition *W* detection by 37% in comparison to the baseline.

### 8.6.3 BLSTM

Table 8.6 *BLSTM results trained on different tasks, using embedded features. BLSTM trained on Task<sub>1</sub> which only using the (read) data, then on Task<sub>2</sub> which using the (read+Spon) data, then on Task<sub>3</sub> which using the (read+Spon+Art) data.*

St	Task <sub>1</sub>			Task <sub>2</sub>			Task <sub>3</sub>		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
I	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
W	1.00	0.05	0.10	0.75	0.08	0.14	0.77	0.29	0.43
P	0.89	0.57	0.70	1.00	0.71	0.83	1.00	0.71	0.83
PH	0.88	0.57	0.69	0.73	0.71	0.72	0.74	0.67	0.70
PW	0.00	0.00	0.00	0.71	0.24	0.36	0.50	0.50	0.50
S	1.00	0.79	0.88	0.93	0.86	0.89	1.00	1.00	1.00
Avg	0.80	0.50	0.56	0.85	0.60	0.66	0.84	0.70	0.74

We trained the BLSTM classifier with word embeddings extracted from pre-trained GloVe model (Common Crawl corpus, 1.9 M vocab). This experiment examined how adding data would affect the performance of the BLSTM classifier. The hyperparameters of the BLSTM were tuned on a development set; the number of hidden nodes is 50, with a word embedding dimension of 300, learning rate and the drop-out rate of 0.001 and 0.5, respectively. All weights in the network were initialized randomly from the uniform distribution within range [-1, 1]. The number of training “epochs” (i.e., iterations) was set to 30. The Tensorflow neural network toolkit (Abadi et al., 2015) was used for BLSTM implementation. The results presented in Table 8.6 show how adding spontaneous and artificial data to the read data improves the results. When only using a read task to train for BLSTM, the performance for detecting stuttering events was very low specially for *W* and *PW*. This is expected due to the small training data set that was used. The increase in the amount of data significantly

improved the performance of the BLSTM classifier because it uses word representation instead of the word entities as learning features. The results obtained from  $CRF_{aux}$  are higher than those obtained from BLSTM. However, BLSTM results without using feature engineering was still considered high and comparable with  $CRF_{aux}$ .

Table 8.7 *Summary table for results on human transcript, using classifiers trained on Task<sub>3</sub>.*

St-type	$CRF_{ngram}$	$CRF_{aux}$	BLSTM
	<b>F1</b>	<b>F1</b>	<b>F1</b>
I	0.88	0.80	1.00
W	0.62	0.99	0.43
P	0.25	1.00	0.83
PH	0.33	0.77	0.70
PW	0.00	0.87	0.50
S	0.96	1.00	1.00
Average	0.49	0.92	0.74

#### 8.6.4 Evaluation on ASR transcripts

The ASR transcripts used in these experiments are transcribed by ASR using the LM augmentation approach described in Chapter five, and they were discussed in our work (Alharbi, Simons, Brumfitt and Green, 2017). As explained in Chapter five, Section 5.2.3, about the LM augmentation approach, this ASR system was created by augmenting a language model with artificially generated stuttering data, which was then able to recognise different stuttering events in the continuous speech of children and also produce a useful word-level transcription of what was said.

The set of experiments presented in the previous sections evaluated the performance of the classifiers on human transcripts (Human\_scoring). However, as mentioned above,

our goal is to detect stuttering in audio recordings directly, bypassing the need for expert transcription. We ask how the proposed models are affected by speech recognition errors. Consequently, the performance of the classifiers needs to be evaluated on ASR transcripts.

These errors propagate to the stuttering detection stage and affect the performance of the classifiers in two ways. First, there is a mismatch between the data used to train the classifiers (human transcripts) and the test set (ASR transcripts), and second the ASR creates different types of errors (i.e. deletion, insertion and substitution errors). The classifiers will label these errors and, even if those labels are correct with respect to the ASR output text, they might be incorrect with respect to the reference transcript. For example, the classifier will label any repeated word that is inserted by ASR as *W* because it is a word repetition. However, those words are inserted by ASR and they do not exist in the reference, which leads to producing a wrong label in comparison to the human reference text, after the alignment process.

In our experiments, the ASR transcripts of the test set have a WER of 12.4% with 1.6% insertion, 3.4% deletion and 7.3% substitution errors. These types of errors affect the stuttering pattern or word fragments detected by the classifiers. The following two examples are taken from the evaluation set and illustrate this argument (Figure 8.3):



for “Human\_scoring”. Moreover, our ASR engine not able to recognize all interjection and prolongation words in the test set (all the 9 interjection and 14 prolongation words were deleted by the ASR), hence *I* and *P* were excluded from the evaluation.

As mentioned before, the aim of scoring classifiers against “Human\_scoring” is to observe how the ASR errors would affect the performance of these classifiers, therefore, we need firstly to score the classifiers on the ASR transcript against “ASR\_scoring” which is the ASR output labelled manually with assumption that WER=0. Scoring the performance of the classifiers against “ASR\_scoring” is straight forward as it has the same length as the ASR transcript.

Table 8.8 Results of classifiers trained on Task<sub>3</sub>, on ASR transcripts (against ASR\_scoring).

St	CRF <sub>ngram</sub>			CRF <sub>aux</sub>			BLSTM		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
W	0.95	0.50	0.66	1.00	1.00	1.00	0.90	0.45	0.60
PH	0.35	0.25	0.29	1.00	0.85	0.92	0.71	0.61	0.65
PW	1.00	0.15	0.26	0.87	0.90	0.88	0.75	0.50	0.60
S	0.94	1.00	0.97	0.94	1.00	0.97	0.94	1.00	0.97
Average	0.81	0.48	0.60	0.95	0.94	0.94	0.82	0.64	0.71

Results demonstrated in Table 8.8 are relatively similar to those reported in the summary table (Table 8.7) on human transcription, using the same classifiers, apart from the drop in the CRF<sub>ngram</sub> and BLSTM classifiers in detecting *PH* events. Since this event is highly context dependent, the change caused by the ASR errors on embedded patterns of word fragments confused both CRF<sub>ngram</sub> and BLSTM classifiers which only depend on word n-gram and word representation features, respectively. However, with the addition of the auxiliary character/utterance based features, as in CRF<sub>aux</sub>, the performance has slightly



improved in detection of *PH* events. As mentioned before, these results mainly reflect the performance of the classifiers on test data and it assumes that WER=0.

To study how classifiers are affected by speech recognition errors and how these errors propagate to the classifiers decisions, the results of the classifiers needs to be compared against reference transcripts (Human\_scoring), which is the actual human transcript.

Table 8.9 Results of classifiers trained on Task<sub>3</sub>, on ASR transcripts (referens<sub>1</sub>).

St-type	CRF <sub>ngram</sub>			CRF <sub>aux</sub>			BLSTM		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
W	0.79	0.47	0.59	0.94	0.91	0.91	0.88	0.44	0.58
PH	0.35	0.23	0.28	0.65	0.73	0.69	0.68	0.57	0.62
PW	1.00	0.12	0.21	1.00	0.55	0.71	0.89	0.40	0.55
S	0.92	0.80	0.86	0.92	0.80	0.86	0.92	0.80	0.86
Average	0.77	0.41	0.53	0.88	0.75	0.81	0.84	0.55	0.66

Due to the ASR insertion and deletion errors, ASR transcripts do not align to “Human\_scoring”. This work follows the alignment procedure described in Liu et al. (2005), in which hypothesized labels are mapped to reference labels using timing information provided by the NIST scoring tool (Kramida et al., 2018). The results evaluating CRF<sub>ngram</sub>, CRF<sub>aux</sub> and BLSTM classifiers on ASR transcription, scored against “Human\_scoring” are shown in Table 8.9. The performances of the three classifiers degrade when applied to ASR, rather than human, transcriptions.

Since there is no previous work, to best of our knowledge, on stuttering detection using automatic recognition output transcripts, we compare our results with those from a similar task (Liu et al., 2005). Following similar alignment method, similar results to those reported in the literature (Baron et al., 2002; Liu et al., 2005) were obtained,

### 8.6.5 Decoding the ‘Arthur the rat’ passage transcribed by ASR using task-oriented lattice approach

Results from previous sections have indicated that, after adding auxiliary features, the  $CRF_{aux}$  classifier gives the best performance. Therefore, this experiment applied the  $CRF_{aux}$ . The training data used in this chapter possibly included the ‘Arthur the rat’ passage, which we used in early experiment to test the ASR. Therefore, this experiment retrains the  $CRF_{aux}$  classifier on  $Task_3$  after removing all recordings for the ‘Arthur the rat’ passage and uses them in the same way as in the test set that was produced from the ASR.

The ASR transcripts in this experiment are transcribed by the task-oriented lattice approach. As explained in Chapter five, Section 5.2.4, this ASR system uses a task-oriented lattice decoding approach to track and identify stuttering events in a reading task. We use a clean original prompt (OP) as an approximation of a manual transcript (which should include stuttering events). This approach can recognise different stuttering events, including sound, part-word, word and phrase repetitions as well as revisions in the continuous speech of children. It can also produce a useful word-level transcription of what was said with an average of 4.2% WER.

As the  $CRF_{aux}$  is not trained to detect and classify a revision event because this would require many examples of revisions in different forms. We, therefore, apply a simple RB algorithm on top of the classifier to capture revision events. Revision occurs at the sentence level but not at the word level. It is most like a phrase repetition; instead of repeating the whole phrase, the child repeats the sentence but changes one or two words. This rule is applied in the RB algorithm to detect revision events.

Table 8.10 Results of  $CRF_{aux}$  trained on  $Task_3$  on ASR transcripts for children who read the ‘Arthur the rat’ passage against “Human\_scoring”, which is the actual labelled human transcript using the annotation approach proposed by Yairi and Ambrose (2005).

St-type	Prec.	Rec.	F1
W	1.00	0.89	0.94
PH	0.97	0.81	0.89
PW	0.92	0.88	0.90
S	0.96	0.92	0.94
R(RB)	0.94	0.90	0.92
Average	0.96	0.88	0.91

Table 8.10 shows the stuttering detection results from the ASR transcript scored against “Human\_scoring”, which is the actual labelled human transcript using the annotation approach proposed by Yairi and Ambrose (2005). The ASR transcripts of this test set have an average WER of 4.2%. As the WER is relatively low the  $CRF_{aux}$  was able to detect and classify an average of 88% of stuttering events, while the average of F-score result indicates a detection of 91% of targeted stuttering events from the transcriptions.

## 8.7 Summary

This chapter evaluates the performance of CRF and BLSTM classifiers for detecting stuttering events in both human and ASR transcripts. The performance variations of the CRF and BLSTM classifiers are analysed by varying the task and the amount of training data.

In the human transcripts, the experimental  $F_1$  results show that, without feature engineering, the BLSTM classifiers outperform the CRF classifiers by 33.6%. However, adding auxiliary features to support the  $CRF_{aux}$  classifier allows performance improvements of 45%

and 18% relative to the CRF baseline ( $CRF_{ngram}$ ) and BLSTM results, respectively on human transcripts.

On the ASR transcripts, the performance of all classifiers degrades after propagating ASR errors. This finding agrees with those reported in the literature for similar tasks. Furthermore, we ascribe this degradation in performance firstly to the ASR errors and secondly to the mismatch between the data used to train the classifiers and the test data. Consequently, future work should focus on reducing ASR errors by reducing the WER and missing rate of stuttering events.

Chapter nine includes the final analysis stage, which is produced by the complete proposed framework by taking the detected stuttering events from ASR transcripts to determine the final severity level of stuttering. It also provides an evaluation of the performance of the framework to diagnose the severity level of stuttering for children who read the ‘Arthur the rat’ passage.

# **Chapter 9**

## **Diagnosing Severity level of Stuttering via a Speech Sample**

### **9.1 Introduction**

The term diagnosis indicates the findings or decisions following assessment and analysis of a disorder or condition. When the nature of the disorder is not precise, an extensive investigation is needed. The professional is responsible for analysing the present signs and indications of the symptoms, which include some objective measures and some subjective opinions to identify the particular nature of the disorder. In some cases, only a single indicator may be adequate, whereas in others, the search for a pattern of symptoms is required. The motivation for the diagnosing process is to facilitate wise judgments concerning the best plan of action to change and improve an unsatisfactory condition or to deal with a disorder.

Patients who stutter state that they diagnose themselves with this disorder during their first visit to a speech language therapist (Yairi and Ambrose, 2005). SLPs never disagree with self-referred adults who diagnosed their speech difficulties as stuttering. When it comes to children who stutter, occasionally, parents have misdiagnosed other communication and speech disorders like stuttering. For example, in one instance parents complained about their

child's stuttering, but SLP diagnosis reported that the problem was hypernasality, which reduced the intelligibility of the child's speech (Yairi and Ambrose, 2005). The important question is whether a child exhibits stuttering symptoms or a normal disfluency.

All samples provided in the UCLASS corpus belonged to children who stutter. However, the severity level of stuttering is not provided for these recorded samples. As mentioned before in previous chapters, our test set was focused on 25 recording samples for children who read the 'Arthur the rat' passage, as this is the most frequent passage read by children in the UCLASS corpus. To determine whether the child in the recording sample exhibits stuttering symptoms or a normal disfluency, analysis and assessment in standard diagnostic methods for stuttering must be undertaken.

This chapter evaluates the ability of our proposed framework to analyse each recording sample and provide an indication of the severity level of stuttering from an acoustic point of view. This evaluation is conducted by comparing our predicted framework severity level results with the results reported by two UK registered SLPs of the same recording samples.

The structure of this chapter as follows. Section 9.2 presents an assessment methodology applied in this chapter; Section 9.3 describes in more detail how to diagnose a speech sample within one of the severity levels; Section 9.4 evaluates the performance of the framework to determine the stuttering severity level for each patient in the test set and 9.5 presents the conclusion and recommendations for future research.

## **9.2 Assessment**

The complete stuttering assessment process is more than analysing a speech sample from a child who stutters. The assessment process includes parent-child interaction and clinician-child interaction sessions which allow the SLP to observe to what extent the child is aware of and reacts emotionally to their stuttering. Assessment also includes a parent interview in order to collect more information about the background history of the family and their

child. We collaborated with a UK registered SLP to create a suitable parent interview form to be used as another feature in our proposed framework in the future (see Appendix A for proposed parent interview form).

From a technological point of view, we can aid the SLP by providing a full-verbatim transcription that includes stuttering events and analysing it. Then, we can indicate the severity level of the stuttering in that recording sample from only the acoustic side. In this thesis, the assessment and diagnosing process regarding the analysis of a speech sample follow the suggestion provided by Guitar (2014). This method is not the only one for analysing a speech sample, however, it is the most recent and comprehensive one, and it takes into account many findings of previous studies as well (Yairi and Ambrose, 2005).

According to Guitar (2014), the recording sample should include at least 200 syllables as that would be enough to observe some stuttering events. Then, the SLP should transcribe the recording, as this transcription could facilitate quantifying the variables of the next stage, called the ‘pattern of disfluencies.’ The ‘pattern of disfluencies’ process includes an analysis of different variables to determine whether the child is truly stuttering. The first variable is the frequency of disfluencies which is computed by extracting the number of disfluencies per 100 words. The second variable is analysing the type of disfluencies, including sound repetition, part word repetition, word repetition, phrase repetition, interjections, revisions, prolongations and blocks. Other variables such as starting/sustaining airflow and physical concomitants cannot be observed with the current proposed ASR approach, as they need additional features to analyse them, which could be addressed in the future. The physical concomitants variable can be observed only if the sample includes a video with audio recordings. This variable includes any physical gesture following a child’s disfluency, such as hand movements, head nods and eye blinks. Starting/sustaining airflow is the observed unexpected onset and offsets of words. Therefore, we assess the first two variables of the test speech recordings by applying the diagnostic approach explained in the next section.

### 9.3 Diagnosis

The purpose of this section is to decide the level of stuttering for children who do stutter. There are three main levels that include: normal disfluency, borderline stuttering and beginning stuttering. The first step is calculating the stuttering-like-disfluencies rate (SLD).

Table 9.1 *This is an example of the approach based on counting disfluencies for sample containing 229 words.*

Disfluency type	Number in sample	Per 100 words
<b><i>Stuttering-Like-Disfluencies (SLD)</i></b>		
Sound repetition	5	2.1
Part-word repetition	1	0.4
Word repetition	2	0.8
Prolongation	5	2.1
<b>Total SLD</b>	13	5.6
<b>Other disfluencies</b>		
Interjection	1	0.4
Revision	1	0.4
Phrase repetition	8	3.4
<b>Total other disfluencies</b>	10	4.3
<b><i>Total disfluencies</i></b>	23	10.0

Table 9.1 shows a detailed example of the disfluency count approach explained by Yairi and Ambrose (2005), to calculate the SLD, which should be applied to diagnose all speech samples in the test set. Their approach considers different types of stuttering disfluency and the number of stuttered words per 100 spoken words in any given speech sample. In



addition, the percentage of SLD detected is critical for the diagnosis process with its focus on the following types of event: sound repetition, part-word repetition, word repetition and dysrhythmic phonation (prolongation). The other disfluencies include interjection, revision and phrase repetition. The third column of Table 9.1 is calculated for each event by

$$(\text{number of stuttering events in sample} * 100) / \text{total words} \quad (9.1)$$

while the SLD percentage of the total number of disfluencies is calculated thusly

$$(\text{total SLD} * 100) / \text{total disfluencies} \quad (9.2)$$

If the percentage of the SLD is less than 50%, then the sample is more likely to be a normal disfluency, whereas if the percentage is more than 50%, then the sample is more likely to be a borderline stuttering. The percentage of SLD of the example in the table is 56.5%. Then, the actual diagnosis process begins by matching specific criteria to one of the following three categories.

*Typical disfluency:* The child must meet all the following criteria to be considered as a child with normal disfluency. They must produce fewer than 10 disfluencies per 100 words. Also, the child cannot have more than one repeated units per repetition, and the rhythm of the repeated unit must be slow and regular. The rate of the SLD to total disfluencies must be <50%. The child must be dealing with these disfluencies normally, if they are aware of them at all.

*Borderline Stuttering; :* The child must meet at least one of the following criteria to be considered a borderline stutterer. The child produces more than 10 disfluencies per 100 words, but they are still relaxed and less tense. They produce two to three repeated units per repetition. The rate of the SLD to total disfluencies is between 50% and 70%.

*Beginning Stuttering:* The key characteristics accompanying stuttering of this level are hurrying and tension. For the child to be considered as a beginning stutterer, they must meet some of the following criteria. Stuttering is rapid, and the child is visibly uncomfortable while speaking. The child repeats more than three repeated units per repetition or suddenly repeats words or phrases. The child is aware of the stuttering and tries to avoid it. There are also problems in starting air flow. Facial expression tense. The rate of the SLD to total disfluencies >70%.

As mentioned in the assessment process, Section 9.2, in our current proposed framework, we are not observing some of the stuttering severity criteria that relate to tension or airflow conditions. Therefore, we focus instead on the characteristics belonging to frequency and type on each stuttering event. Table 9.2 shows the applied criteria that could be observed in our current proposed framework for each severity level.

Table 9.2 *Stuttering severity levels.*

Category	Characteristics
Normal disfluency	No greater than 10 disfluencies per 100 words. One-unit repetitions. Most common disfluency types are interjections, revisions, and word repetitions. The rate of the SLD to total disfluencies <50%.
Borderline stutter	More than 10 disfluencies events per 100 words. Two to three-unit repetitions. The rate of the SLD to total disfluencies between 50% and 70%. More repetitions and prolongations than revisions or interjections.
Beginning stutter	More than three-unit repetitions. More prolongations and repetitions. The rate of the SLD to total disfluencies >70%.

## 9.4 Framework evaluation

The detected stuttering events using the proposed framework are evaluated to determine the severity level of stuttering for each recording sample in the test sets. The test includes 25 sample recordings of children reading the ‘Arthur the rat’ passage from UCLASS Release Two corpus (Howell et al., 2009). The number of detected events after analysing the transcripts produced by the ASR, in conjunction with identified prolongation using the diagnostic approach to stuttering described in Section 9.3, are assessed. Then, each speech sample is classified as normal disfluency, borderline stutter or beginning stutter.

Additionally, each sample is also diagnosed by two UK registered SLPs, and the severity of stuttering is then determined using the same approach described in Section 9.3. Then, a comparison between results obtained by the proposed system and the results obtained by the SLPs is performed.

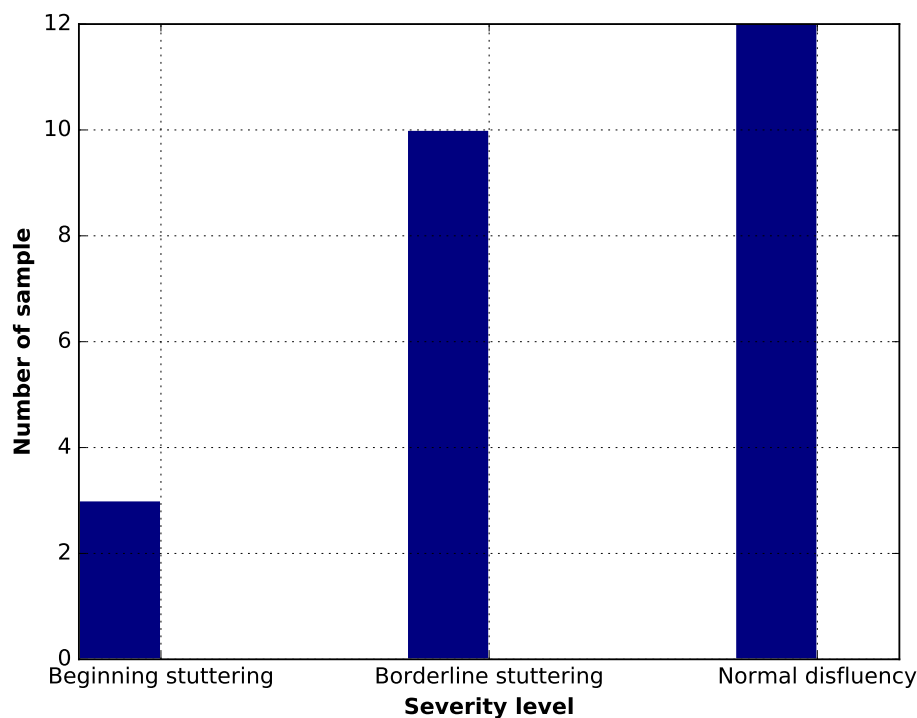


Figure 9.1 *Ground truth diagnosis recordings sample.*

Figure 9.1 demonstrates the number of samples belonging to each category according to the diagnose provided by the SLPs.

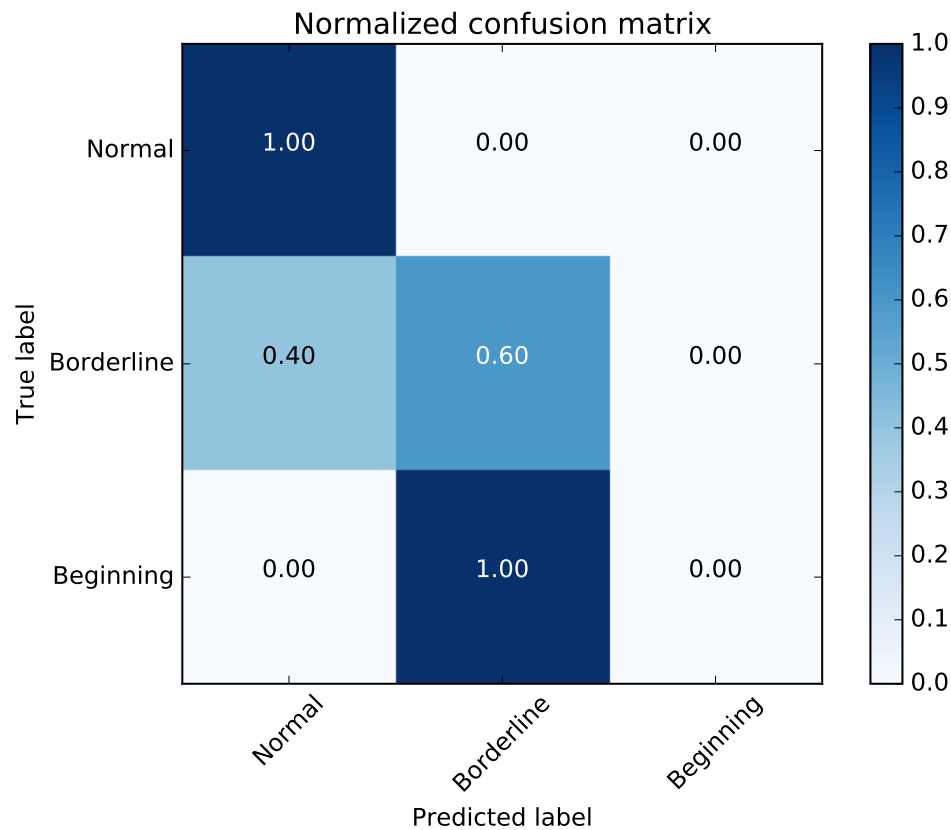


Figure 9.2 *Confusion matrix of the predicted severity level from proposed framework with the actual severity level diagnosed by a SLP.*

As shown in Figure 9.2, the results from the proposed framework are compared with those obtained from the SLPs. The comparison demonstrates that our framework can correctly classify the level of stuttering severity in 72% of the test samples. The figure shows that most of the incorrectly classified samples are in a neighbouring category. The incorrectly classified borderline samples are classified as normal disfluency while all beginning stuttering samples are classified as borderline samples. This is mostly because the ASR underreports some stuttering events which affect the final diagnosing process. The ASR misses about 33.6% of stuttering events in the actual audio and the stuttering events detector underreports 9% of

stuttering events in the transcription produced by the ASR. Therefore, as Yairi and Ambrose (2005) specify in their SLD diagnostic method, calling for 50% of borderline stuttering class and 70% for beginning stuttering class, we minimise this threshold for both classes to 30% and 50%, respectively, to allow the system to cover the underreported events. By minimising the diagnostic threshold, all missed neighbouring classes fit into the correct pattern.

After tuning our diagnostic system to accommodate the errors of the ASR, we were able to classify 100% of the test subjects as belonging to the correct stuttering category, with regards to severity. However, even the obtained result without adjusting the diagnostic threshold is promising, as it indicates that further future tuning in the framework may improve performance. For instance, adding more features to the framework, such as historical background and video analysis, will enhance the diagnosis decision.

## 9.5 Summary

This chapter evaluates the final performance of our proposed automatic framework for diagnosing the severity level of children who stutter. The performance of the framework is compared with the diagnosis on similar test recording samples by registered UK SLPs. The results indicate that the framework can correctly classify the level of stuttering severity in 72% of the test samples. Most incorrectly-classified samples are in a neighbouring category. By applying a suggestion that a diagnostic threshold could be minimised to cover the missed stuttering events which helps to fit all missed neighbour classes into the correct one.



# Chapter 10

## Conclusion

Therapists traditionally count up patient's speech disfluencies to determine the severity of his or her stuttering. In a more accurate approach, they record a clinical session and transcribe each spoken word in the recorded speech sample offline to determine the severity level of stuttering. However, subjective stuttering evaluation is time consuming and extremely dependent on the experience of the therapist. Therefore, an automated transcription tool of a given recording sample that can analyse it and provide an initial diagnosis of stuttering severity would be useful for therapists.

This thesis has investigated the possibility of building an automated framework to aid therapists in diagnosing children who stutter. It was concerned with the generation of an automatic diagnosing of stuttering for children from a speech sample and full-verbatim transcription of uploaded recording speech, which could be used later to monitor the child's progress and for further research investigations. Initially, time-aligned transcriptions including stuttering events were created for a complete UCLASS, Release Two dataset of read speech, provided by (Howell et al., 2009), then a word-level annotation for relevant stuttering categories for each word was generated (see Chapter 4). Next, the ASR was conducted using two different approaches. The first approach was an ASR trained acoustically on the PF-Star corpus and Howell's UCLASS data, with the language model (LM) artificially augmented

with pseudo-words and repetitions to increase the frequency of stuttering events (sound, part-word and whole-word repetition; phrase repetition and revision). Then, a task-specific lattices was developed from the original reading prompts to build more constrained and specified LM that allowed for several stuttering events (see Chapter 5 and 6). In parallel, the prolongation detector processing the speech sample to detect prolongation events using proposed unsupervised approach to detect prolongation segments without any need to train a classifier (see Chapter 7). The automatic transcriptions for each recording sample were analysed using two machine learning approaches to detect and classify stuttering events and submit them to the diagnosing system (see Chapter 8). Finally, to verify the efficiency of the automatic diagnostic framework, the frequency and type of each stuttering events provided by automatic transcriptions and the prolongation detector were compared and evaluated using human judgment evaluation, provided by two UK registered SLPs (see Chapter 9).

## 10.1 Research findings

This section provides a summary of the main research findings discovered during this period of research, whether or not they were immediately relevant to our project goals.

- **Corpus Generation and Analysis.**

- In order to develop an adapted ASR system that would be able to recognise frequency-based stuttering events (including sound, part-word, word and phrase repetitions, as well as revision), there was a need for a new full-verbatim, time-aligned transcription for read recording samples of the UCLASS Release Two corpus.
- To train different classifiers to detect stuttering events from transcriptions produced by the ASR, there was a need for a word-level annotation.



- A segmentation process was applied to a subset of read recording samples from the UCLASS Release Two corpus in order to experiment with supervised and unsupervised approaches in detecting prolongation events from a speech signal by identifying prolonged and non-prolonged segments.
  - In order to develop a full diagnostic method, there is a need to capture information from parents. In cooperation with a SLP, we were able to create a reliable questionnaire for parents that can capture different important aspects such as family history, contextual hereditary and stress factors that could impact stuttering (see Appendix A).
- **Adapted ASR system to detect stuttering events using two different approaches.**
    - The LM of the first ASR approach was artificially augmented with pseudo-words and repetitions to increase and detect the frequency of those stuttering events mentioned above. The average improvement in detecting stuttering events, as evidenced by the lower miss rate, is 48.13%.
    - The second ASR approach was a task-oriented lattices developed from the original reading prompts to build a more constrained and specified LM that allowed for different stuttering events, including sound, part-word, word, phrase repetitions and revision. Using this approach, we were able to obtain an improvement by lower the miss rate to 33.60%.
    - The task-oriented lattice decoding approach outperformed the LM augmentation approach, with the difference considered statistically significant. The system preserves an average of 66.40% of the stuttering events after applying the task-oriented lattices approach, while only 51.87% of stuttering events were preserved after applying the LM augmentation.

- The word error rate (WER) may not be a particularly effective statistic for detecting stuttering, as substitutions and insertions in normal word (not stuttered word) clearly affect the WER, but this happens independently of the existence of stuttering events.

- **Prolongation detection system**

- Attempting to apply a supervised approach, which uses machine-learning classifiers such as support vector machine (SVM) and k-nearest neighbours (KNN) to classify prolongation segments directly from a speech signal, failed with a 9% F1 for KNN and 17% F1 for SVM due to the limited data used. A further issue relates to data balance. Therefore, it is difficult to generalise learning across stuttering events that are phonetically distinct. When counting a stuttering event, prolongation is just one event for any prolonged phone. However, as the utterance may involve different prolonged phones such as mmm or ffff, there is a need for more training data in order to classify all of these as one prolongation event. This gave us an indication that the direct learning approach was not going to work properly in our diagnostic system with the current limited data and other techniques should be applied.
- Our second attempt of the supervised approach proposed different solutions to the unbalanced data problem and improved precision (88%) and recall (58%) by applying a UBM-based approach and trying to detect a prolongation event rather than classifying it by applying a moving window through a test signal. However, the supervised approach appeared inadequate, as 58% recall is insufficient for diagnostic purposes.
- The unsupervised approach is based on the autocorrelation function (ACF), which measures the similarity between successive speech frames and applies several

automatic-based thresholds to filter the duration of each detected segment. This approach improved recall to 92%, but precision reduced to 60%.

- **Analysing transcripts produced by the ASR using machine learning classifiers.**
  - Training machine-learning classifiers, such as conditional random fields (CRF) and bi-directional long-short-term memory (BLSTM), to detect and classify stuttering events from transcription is possible after retraining them with augmented data. The best results achieved was 92% for F1 after applying CRF on augmented data and trained with auxiliary features such as adding character- and utterance-based features.
- **Diagnose severity level of stuttering of a speech sample.**
  - The evaluation of the final framework's performance was compared with the diagnosis on similar test recording samples by registered UK SLPs, which confirmed that the framework can correctly classified the level of stuttering severity in 72% of the test samples. Most incorrectly-classified samples were in a neighbouring category because the framework was under-reporting the stuttering events. By tuning the diagnostic threshold, the system is able to fit all missed neighbour classes into the correct one.

## 10.2 In support of the thesis

- RQ1: *Given that there is scarce stuttering data available to train an ASR, which is the best ASR approach for detecting stuttering events?*

We tried the augmented LM approach which gave some improvement over the baseline to reach a miss rate of 48.13%. We also tried the task-oriented lattice approach where we introduced different stuttering arcs on a lattice derived from the original prompt

with 33.60% of miss rate. It turned out that the task-oriented lattice approach was 23.27% more successful in recognising the detailed sounds, words, part-word, phrase repetitions and revisions that would be necessary to feed forward a stuttering events classifier.

- **RQ2: *How do we reliably detect time-based stuttering events when we do not have enough data, and what would the best techniques be?***

The main time-based events in stuttering are prolongation and blocks events. From our findings, we found that eventually we were able to build a robust and fully automatic prolongation detection system based on autocorrelation function to estimate the similarity between successive speech frames and proposed prolongation events in continuous speech. However, we were not able to detect blocking because it was difficult to distinguish blocking from ordinary silences.

- **RQ3: *What is the best approach for detecting stuttering events from transcription?***

The hand-written, rule-based approach could be one solution here to detect stuttering events from transcription. However, this approach depends on the expert's knowledge (Liu et al., 2016), which means it only works if all situations of stuttering events are considered. This condition cannot be satisfied in practice due to the continuous variability in data volume and complexity. Therefore, applying a statistical classifier over a dataset that can be augmented and then retrained may, in the long run, have been the better approach. From the findings it appears that the best result achieved was 92% for F1 by applying a CRF classifier trained with auxiliary features such as adding character- and utterance-based features on augmented data.

- RQ4: *Can we reliably diagnose the severity of a child's stuttering by automatic means?*

Depending on the recorded speech samples, we could provide a strong indication of the severity level of stuttering. The proposed framework was able to give the SLP an indication of the severity level of stuttering in the applied test set (25 recorded speeches of the 'Arthur the rat' passage). The proposed system correctly classified 72% of the tested recordings and 100% after tuning the diagnostic threshold. This indication is purely from recorded audio, without depending on any further metadata, which will be helpful for a fuller diagnosis method.

We created an 'interview form', which is an interview for a patient, done with the purpose of trying to discover contextual hereditary and stress factors behind the stuttering that are potentially making it worse (see Appendix A). Unfortunately, we were not able to automatically include the interview in the diagnostic process because there was no chance to contact the parents of the children in the applied recordings.

### 10.2.1 Original contributions

1. **Corpus Generation and Analysis.** In order to develop an adapted ASR system able to recognise frequency-based stuttering events (including sound, part-word, word and phrase repetitions as well as revision) a new full-verbatim and time-aligned transcriptions for all read recordings sample of UCLASS Release Two corpus was generated. We also created a word-level annotation to train different classifiers to detect stuttering events from transcriptions produced by the ASR. The transcription and annotation approaches was reviewed by a UK registered speech language therapist.
2. **Adapted ASR system to detect stuttering events using two different approaches.** This contribution proposed two main adapted ASR approaches to detect different

stuttering events. Both approaches were based on building an ASR that focused acoustically on two main datasets. The first dataset was a small stuttered corpus from UCLASS (Howell et al., 2009), and the second was a larger corpus of clean children's speech provided by the PF-Star dataset (Russell, 2006). The LM of the first approach was artificially augmented with pseudo-words and repetitions to increase and detect the frequency of those stuttering events mentioned above. The second approach used a task-specific lattices developed from the original reading prompts to build a more constrained and specified LM that allowed for different stuttering events including sound, part-word, word, phrase repetitions as well as revision.

3. **Prolongation detection system** This system was developed to detect prolongation events. This system proposes a refined workflow that aims to detect and preserve time and count information of the prolongation event. To address this problem, a fully automatic system based on autocorrelation function (ACF) was implemented to estimate the similarity between successive speech frames and proposed prolongation events. A silence remover is applied to minimise the false alarms of detected similar segments. Moreover, a speaking rate estimation with a maximum tolerance of one second was applied as a threshold to filter detected similar segments by ACF.
4. **Analysing transcripts produced by the ASR using machine learning classifiers.** Two machine-learning approaches—conditional random fields (CRF) and bi-directional long-short-term memory (BLSTM)—were investigated to detect stuttering events from both human and ASR transcripts of children's read speech and compare the performance of each. The comparison was conducted using lexical, contextual and geometrical features. In addition, the study was conducted on the effect of adding more data to the performance of the classifiers. This study also investigated the effect of augmenting the available training data with artificially generated data in order

to improve the classifiers' performance. Finally, this work describes a method for studying the effect of ASR errors on classifiers' performances.

5. **Diagnosing severity level of stuttering of a speech sample.** All frequency and types of detected stuttering events were collected and analysed by a classifier inspired by the Guitar (2014) diagnosis method. This classifier is responsible for designating stutters into three main categories based on the types and count of stuttering events: normal disfluency, borderline stuttering or beginning stuttering.

## 10.3 Future work

The previous section describes the summary of the main findings and contributions in this thesis. There is, however, some room for improvement to enhance the overall performance of the proposed framework. Future directions for this work are explored below.

1. Adding more metadata to the frequency and type of stuttering events by extracting different information from our produced questionnaire 'interview\_form' (see Appendix A), such as age, sex, family history and left hand writing, and use them as extra features to train a classifier to automatically include them in the diagnostic process (Geetha et al., 2000).
2. One necessity that must be met is the collection of more data on children who stutter. One idea to accomplish this is launching the proposed framework into the cloud to be considered as a cloud-based system. Then, this framework would facilitate the collection of more data on the disordered speech of affected patients through registered therapists. Additionally, a speech therapist who is counselling a patient might not have appropriate expertise. It may be possible to provide access to appropriate similar cases through the cloud and thereby improve diagnosis through data sharing.





# Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y. and Zheng, X. (2015), ‘TensorFlow: Large-scale machine learning on heterogeneous systems’. Software available from tensorflow.org.

**URL:** <https://www.tensorflow.org/>

Abercrombie, D. (1964), *English Phonetic Texts.(1. Publ.)*, Faber & Faber.

Ai, O. C., Hariharan, M., Yaacob, S. and Chee, L. S. (2012), ‘Classification of speech dysfluencies with mfcc and lpcc features’, *Expert Systems with Applications* **39**(2), 2157–2165.

Alharbi, S., Hasan, M., Simons, A.-J., Brumfitt, S. and Green, P. (2017), Detecting stuttering events in transcripts of children’s speech, in ‘International Conference on Statistical Language and Speech Processing’, Springer, pp. 217–228.

Alharbi, S., Hasan, M., Simons, A.-J., Brumfitt, S. and Green, P. (2018), ‘A lightly supervised approach to detect stuttering in children’s speech’, *Proc. Interspeech 2018* pp. 3433–3437.

Alharbi, S., Simons, A. J., Brumfitt, S. and Green, P. D. (2017), Automatic recognition of children’s read speech for stuttering application, in ‘6th. Workshop on Child Computer

- Interaction (WOCCI 2017), eds. K. Evanini, M. Najafian, S. Safavi and K. Berkling', International Speech Communication Association (ISCA), pp. 1–6.
- Ambrose, N. G., Cox, N. J. and Yairi, E. (1997), 'The genetic basis of persistence and recovery in stuttering', *Journal of Speech, Language, and Hearing Research* **40**(3), 567–580.
- Andrade, C. R. F. d., Cervone, L. M. and Sassi, F. C. (2003), 'Relationship between the stuttering severity index and speech rate', *Sao Paulo Medical Journal* **121**(2), 81–84.
- Andrews, G., Hoddinott, S., Craig, A., Howie, P., Feyer, A.-M. and Neilson, M. (1983), 'Stuttering: a review of research findings and theories circa 1982', *Journal of speech and hearing disorders* **48**(3), 226–246.
- Bahl, L. R., Jelinek, F. and Mercer, R. L. (1983), 'A maximum likelihood approach to continuous speech recognition', *IEEE Transactions on Pattern Analysis & Machine Intelligence* (2), 179–190.
- Balakrishnama, S. and Ganapathiraju, A. (1998), 'Linear discriminant analysis-a brief tutorial', *Institute for Signal and information Processing* **18**.
- Banerjee, S., Beck, J. E. and Mostow, J. (2003), Evaluating the effect of predicting oral reading miscues, in 'Eighth European Conference on Speech Communication and Technology'.
- Banerjee, S., Mostow, J., Beck, J. and Tam, W. (2003), Improving language models by learning from speech recognition errors in a reading tutor that listens, in 'Proceedings of the Second International Conference on Applied Artificial Intelligence', Citeseer.
- Baron, D., Shriberg, E. and Stolcke, A. (2002), Automatic punctuation and disfluency detection in multi-party meetings using prosodic and lexical cues, in 'Seventh International Conference on Spoken Language Processing'.

- Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970), 'A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains', *The annals of mathematical statistics* **41**(1), 164–171.
- Bellegarda, J. R. (2004), 'Statistical language model adaptation: review and perspectives', *Speech communication* **42**(1), 93–108.
- Bengio, Y., Simard, P. and Frasconi, P. (1994), 'Learning long-term dependencies with gradient descent is difficult', *IEEE transactions on neural networks* **5**(2), 157–166.
- Bisani, M. and Ney, H. (2008), 'Joint-sequence models for grapheme-to-phoneme conversion', *Speech communication* **50**(5), 434–451.
- Blood, G. W. and Blood, I. M. (2004), 'Bullying in adolescents who stutter: Communicative competence and self-esteem', *Contemporary Issues in Communication Science and Disorders* **31**, 69–79.
- Blood, G. W. and Blood, I. M. (2007), 'Preliminary study of self-reported experience of physical aggression and bullying of boys who stutter: Relation to increased anxiety', *Perceptual and motor skills* **104**(3\_suppl), 1060–1066.
- Blood, G. W., Blood, I. M., Tellis, G. and Gabel, R. (2001), 'Communication apprehension and self-perceived communication competence in adolescents who stutter', *Journal of Fluency Disorders* **26**(3), 161–178.
- Blood, G. W., Mamett, C., Gordon, R. and Blood, I. M. (2010), 'Written language disorders: Speech-language pathologists' training, knowledge, and confidence', *Language, Speech, and Hearing Services in Schools* **41**(4), 416–428.
- Bloodstein, O. (1969), 'A handbook on stuttering.'
- Blumgart, E., Tran, Y. and Craig, A. (2010), 'Social anxiety disorder in adults who stutter', *Depression and Anxiety* **27**(7), 687–692.

- Bourlard, H. A. and Morgan, N. (2012), *Connectionist speech recognition: a hybrid approach*, Vol. 247, Springer Science & Business Media.
- Box, J. (1994), 'Reinsel, editor. time series analysis, forecasting and control. englewood clifs'.
- Brisk, D. J., Healey, E. C. and Hux, K. A. (1997), 'Clinicians' training and confidence associated with treating school-age children who stutter: A national survey', *Language, Speech, and Hearing Services in Schools* **28**(2), 164–176.
- Brundage, S. B., Bothe, A. K., Lengeling, A. N. and Evans, J. J. (2006), 'Comparing judgments of stuttering made by students, clinicians, and highly experienced judges', *Journal of Fluency Disorders* **31**(4), 271–283.
- Campbell, W. M., Sturim, D. E. and Reynolds, D. A. (2006), 'Support vector machines using gmm supervectors for speaker verification', *IEEE signal processing letters* **13**(5), 308–311.
- Cevikalp, H. (2017), 'Best fitting hyperplanes for classification', *IEEE transactions on pattern analysis and machine intelligence* **39**(6), 1076–1088.
- Chan, H. Y. and Woodland, P. (2004), Improving broadcast news transcription by lightly supervised discriminative training, in 'Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on', Vol. 1, IEEE, pp. I–737.
- Chandrakala, S. and Rajeswari, N. (2017), 'Representation learning based speech assistive system for persons with dysarthria', *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **25**(9), 1510–1517.
- Chee, L. S., Ai, O. C., Hariharan, M. and Yaacob, S. (2009), Mfcc based recognition of repetitions and prolongations in stuttered speech using k-nn and lda, in 'Research and Development (SCORED), 2009 IEEE Student Conference on', IEEE, pp. 146–149.
- Chen, L., Lamel, L. and Gauvain, J. L. (2004), Lightly supervised acoustic model training using consensus networks, in '2004 IEEE International Conference on Acoustics, Speech, and Signal Processing', Vol. 1, pp. I–189–92 vol.1.

- Chen, S. F. and Goodman, J. (1999), 'An empirical study of smoothing techniques for language modeling', *Computer Speech & Language* **13**(4), 359–394.
- Chiu, J. P. and Nichols, E. (2015), 'Named entity recognition with bidirectional lstm-cnns', *arXiv preprint arXiv:1511.08308* .
- Claes, T., Dologlou, I., ten Bosch, L. and Van Compernelle, D. (1998), 'A novel feature transformation for vocal tract length normalization in automatic speech recognition', *IEEE Transactions on Speech and Audio Processing* **6**(6), 549–557.
- CMU (1998), 'The carnegie mellon university (cmu) american english pronunciation dictionary', *Carnegie Mellon University*, Web: <http://www.speech.cs.cmu.edu/cgi-bin/cmudict> .
- Conture, E. G. (2001), *Stuttering: Its nature, diagnosis, and treatment*, Allyn & Bacon.
- Cook, S., Donlan, C. and Howell, P. (2013), 'Stuttering severity, psychosocial impact and lexical diversity as predictors of outcome for treatment of stuttering', *Journal of fluency disorders* **38**(2), 124–133.
- Cook, S. and Howell, P. (2013), 'Children's and parents' perspectives of psychosocial impact of stuttering and stuttering-related bullying'.
- Craig, A. (1998), *Treating stuttering in older children, adolescents and adults: a guide for clinicians, parents and those who stutter*, Feedback publications.
- Craig, A., Blumgart, E. and Tran, Y. (2009), 'The impact of stuttering on the quality of life in adults who stutter', *Journal of fluency disorders* **34**(2), 61–71.
- Craig, A. and Calver, P. (1991), 'Following up on treated stutterers studies of perceptions of fluency and job status', *Journal of Speech, Language, and Hearing Research* **34**(2), 279–284.
- Craig, A. and Tran, Y. (2006), 'Fear of speaking: chronic anxiety and stammering', *Advances in Psychiatric Treatment* **12**(1), 63–68.

- Czyzewski, A., Kaczmarek, A. and Kostek, B. (2003), 'Intelligent processing of stuttered speech', *Journal of Intelligent Information Systems* **21**(2), 143–171.
- Dahl, G. E., Yu, D., Deng, L. and Acero, A. (2012), 'Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition', *IEEE Transactions on audio, speech, and language processing* **20**(1), 30–42.
- Davis, S. and Mermelstein, P. (1980), 'Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences', *IEEE transactions on acoustics, speech, and signal processing* **28**(4), 357–366.
- de Jong, N. H., Wempe, T. et al. (2007), Automatic measurement of speech rate in spoken dutch, in 'ACLIC Working Papers', Vol. 2, Citeseer, pp. 51–60.
- Dehak, N., Dehak, R., Kenny, P., Brümmer, N., Ouellet, P. and Dumouchel, P. (2009), Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification, in 'Tenth Annual conference of the international speech communication association'.
- Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P. and Ouellet, P. (2011), 'Front-end factor analysis for speaker verification', *IEEE Transactions on Audio, Speech, and Language Processing* **19**(4), 788–798.
- Dehak, N., Torres-Carrasquillo, P. A., Reynolds, D. and Dehak, R. (2011), Language recognition via i-vectors and dimensionality reduction, in 'Twelfth annual conference of the international speech communication association'.
- Duenser, A., Ward, L., Stefani, A., Smith, D., Freyne, J., Morgan, A. and Dodd, B. (2016), Feasibility of technology enabled speech disorder screening, in 'Digital Health Innovation for Consumers, Clinicians, Connectivity and Community: Selected Papers from the 24th Australian National Health Informatics Conference (HIC 2016)', Vol. 227, IOS Press, p. 21.

- Echeverry-Correa, J. D., Ferreiros-López, J., Coucheiro-Limeres, A., Córdoba, R. and Montero, J. M. (2015), 'Topic identification techniques applied to dynamic language model adaptation for automatic speech recognition', *Expert Systems with Applications* **42**(1), 101–112.
- Elenius, D. and Blomberg, M. (2004), 'Comparing speech recognition for adults and children', *Proceedings of FONETIK 2004* pp. 156–159.
- Esmaili, I., Dabanloo, N. J. and Vali, M. (2017), 'An automatic prolongation detection approach in continuous speech with robustness against speaking rate variations', *Journal of medical signals and sensors* **7**(1), 1.
- Federico, M., De Mori, R. and Ponting, K. (1999), 'Language model adaptation'.
- Fiscus, J. G., Ajot, J., Garofolo, J. S. and Doddington, G. (2007), Results of the 2006 spoken term detection evaluation, in 'Proc. sigir', Vol. 7, pp. 51–57.
- Fleiss, J. L. and Cohen, J. (1973), 'The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability', *Educational and psychological measurement* **33**(3), 613–619.
- Forney, G. D. (1973), 'The viterbi algorithm', *Proceedings of the IEEE* **61**(3), 268–278.
- Fredouille, C., Pouchoulin, G., Bonastre, J.-F., Azzarello, M., Giovanni, A. and Ghio, A. (2005), Application of automatic speaker recognition techniques to pathological voice assessment (dysphonia), in 'Proceedings of European Conference on Speech Communication and Technology (Eurospeech 2005)', ISCA, pp. 149–152.
- Fringi, E., Lehman, J. F. and Russell, M. (2015), Evidence of phonological processes in automatic recognition of children's speech, in 'Sixteenth Annual Conference of the International Speech Communication Association'.
- Geetha, Y., Pratibha, K., Ashok, R. and Ravindra, S. K. (2000), 'Classification of childhood disfluencies using neural networks', *Journal of fluency disorders* **25**(2), 99–117.

- Gers, F. A., Schraudolph, N. N. and Schmidhuber, J. (2002), 'Learning precise timing with lstm recurrent networks', *Journal of machine learning research* **3**(Aug), 115–143.
- Gillick, L. and Cox, S. J. (1989), Some statistical issues in the comparison of speech recognition algorithms, in 'International Conference on Acoustics, Speech, and Signal Processing', IEEE, pp. 532–535.
- Graves, A., Mohamed, A.-r. and Hinton, G. (2013), Speech recognition with deep recurrent neural networks, in 'Acoustics, speech and signal processing (icassp), 2013 ieee international conference on', IEEE, pp. 6645–6649.
- Gregory, H. H., Campbell, J. H., Gregory, C. B. and Hill, D. G. (2003), *Stuttering therapy: Rationale and procedures*, Allyn & Bacon.
- Guiter, B. (2014), *Stuttering: An integrated approach to its nature and treatment*, Lippincott Williams & Wilkins.
- Hämäläinen, A., Candeias, S., Cho, H., Meinedo, H., Abad, A., Pellegrini, T., Tjalve, M., Trancoso, I. and Dias, M. S. (2014), Correlating asr errors with developmental changes in speech production: A study of 3-10-year-old european portuguese children's speech, in 'Workshop on Child Computer Interaction-WOCCI 2014', pp. pp–1.
- Hayhow, R., Cray, A. M. and Enderby, P. (2002), 'Stammering and therapy views of people who stammer', *Journal of Fluency disorders* **27**(1), 1–17.
- Haykin, S. (1999), Neural networks: A comprehensive foundation, in '2nd ed., Prentice–Hall, Inc., Englewood Cliffs, NJ', Prentice–Hall.
- Hazen, T. J. (2006), Automatic alignment and error correction of human generated transcripts for long speech recordings, in 'Ninth International Conference on Spoken Language Processing'.
- Hearne, A., Packman, A., Onslow, M. and Quine, S. (2008), 'Stuttering and its treatment in adolescence: The perceptions of people who stutter', *Journal of Fluency Disorders* **33**(2), 81–98.



- Heeman, P. A., Lunsford, R., McMillin, A. and Yaruss, J. S. (2016), 'Using clinician annotations to improve automatic speech recognition of stuttered speech', *Interspeech 2016* pp. 2651–2655.
- Heeman, P. A., McMillin, A. and Yaruss, J. S. (2011), Computer-assisted disfluency counts for stuttered speech., *in* 'INTER\_SPEECH', pp. 3013–3016.
- Hermansky, H. (1990), 'Perceptual linear predictive (plp) analysis of speech', *the Journal of the Acoustical Society of America* **87**(4), 1738–1752.
- Hermansky, H., Ellis, D. P. and Sharma, S. (2000), Tandem connectionist feature extraction for conventional hmm systems, *in* '2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)', Vol. 3, IEEE, pp. 1635–1638.
- Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Kingsbury, B. et al. (2012), 'Deep neural networks for acoustic modeling in speech recognition', *IEEE Signal processing magazine* **29**.
- Hochreiter, S. and Schmidhuber, J. (1997), 'Long short-term memory', *Neural computation* **9**(8), 1735–1780.
- Hollingshead, K. and Heeman, P. (2004), 'Using a uniform-weight grammar to model disfluencies in stuttered read speech: a pilot study', *Center for Spoken Language Understanding* pp. 1–22.
- Honal, M. and Schultz, T. (2003), Correction of disfluencies in spontaneous speech using a noisy-channel approach, *in* 'Eighth European Conference on Speech Communication and Technology'.
- Honal, M. and Schultz, T. (2005), Automatic disfluency removal on recognized spontaneous speech-rapid adaptation to speaker-dependent disfluencies, *in* 'Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on', Vol. 1, IEEE, pp. I–969.

- Howell, P. (2011), *Recovery from stuttering*, Routledge.
- Howell, P. (2013), 'Screening school-aged children for risk of stuttering', *Journal of fluency disorders* **38**(2), 102–123.
- Howell, P. and Davis, S. (2011), 'Predicting persistence of and recovery from stuttering by the teenage years based on information gathered at age 8 years', *Journal of Developmental & Behavioral Pediatrics* **32**(3), 196–205.
- Howell, P., Davis, S. and Bartrip, J. (2009), 'The university college london archive of stuttered speech (uclass)', *Journal of Speech, Language, and Hearing Research* **52**(2), 556–569.
- Howell, P. and Sackin, S. (1995), Automatic recognition of repetitions and prolongations in stuttered speech, in 'Proceedings of the first World Congress on fluency disorders', Vol. 2, pp. 372–374.
- Howell, P., Sackin, S. and Glenn, K. (1997), 'Development of a two-stage procedure for the automatic recognition of dysfluencies in the speech of children who stutter: Ii. ann recognition of repetitions and prolongations with supplied word segment markers', *Journal of Speech, Language, and Hearing Research* **40**(5), 1085–1096.
- Howell, P. and Vause, L. (1986), 'Acoustic analysis and perception of vowels in stuttered speech', *The Journal of the Acoustical Society of America* **79**(5), 1571–1579.
- Hu, Z., Ma, X., Liu, Z., Hovy, E. and Xing, E. (2016), 'Harnessing deep neural networks with logic rules', *arXiv preprint arXiv:1603.06318* .
- Huang, Z., Xu, W. and Yu, K. (2015), 'Bidirectional lstm-crf models for sequence tagging', *arXiv preprint arXiv:1508.01991* .
- Ingham, R. J. and Cordes, A. K. (1998), 'Treatment decisions for young children who stutter: Further concerns and complexities', *American Journal of Speech-Language Pathology* **7**(3), 10.

- Iverach, L., O'Brian, S., Jones, M., Block, S., Lincoln, M., Harrison, E., Hewat, S., Menzies, R. G., Packman, A. and Onslow, M. (2009), 'Prevalence of anxiety disorders among adults seeking speech therapy for stuttering', *Journal of anxiety disorders* **23**(7), 928–934.
- Jelinek, F. (1997), *Statistical methods for speech recognition*, MIT press.
- Jelinek, F., Mercer, R. L., Bahl, L. R. and Baker, J. K. (1977), 'Perplexity—a measure of the difficulty of speech recognition tasks', *The Journal of the Acoustical Society of America* **62**(S1), S63–S63.
- Jiang, J., Lu, C., Peng, D., Zhu, C. and Howell, P. (2012), 'Classification of types of stuttering symptoms based on brain activity', *PloS one* **7**(6), e39747.
- John, G. and Holliman, E. (1993), 'Switchboard-1 release 2 ldc97s62', *Web Download. Philadelphia: Linguistic Data Consortium* .
- Johnson, W. (1961), 'Measurements of oral reading and speaking rate and disfluency of adult male and female stutterers and nonstutterers.', *The Journal of speech and hearing disorders* p. 1.
- Jones, M., Onslow, M., Packman, A., Williams, S., Ormond, T., Schwarz, I. and Gebiski, V. (2005), 'Randomised controlled trial of the lidcombe programme of early stuttering intervention', *BMJ* **331**(7518), 659.  
**URL:** <http://www.bmj.com/content/331/7518/659>
- Jones, R. M., Walden, T. A., Conture, E. G., Erdemir, A., Lambert, W. E. and Porges, S. W. (2017), 'Executive functions impact the relation between respiratory sinus arrhythmia and frequency of stuttering in young children who do and do not stutter', *Journal of Speech, Language, and Hearing Research* **60**(8), 2133–2150.
- Kawahara, T., Nemoto, Y. and Akita, Y. (2008), Automatic lecture transcription by exploiting presentation slide information for language model adaptation, *in* 'Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on', IEEE, pp. 4929–4932.

- Kenny, P., Boulianne, G. and Dumouchel, P. (2005), 'Eigenvoice modeling with sparse training data', *IEEE transactions on speech and audio processing* **13**(3), 345–354.
- Kinnunen, T. and Li, H. (2010), 'An overview of text-independent speaker recognition: From features to supervectors', *Speech communication* **52**(1), 12–40.
- Klein, J. F. and Hood, S. B. (2004), 'The impact of stuttering on employment opportunities and job performance', *Journal of fluency disorders* **29**(4), 255–273.
- Klompas, M. and Ross, E. (2004), 'Life experiences of people who stutter, and the perceived impact of stuttering on quality of life: personal accounts of south african individuals', *Journal of fluency disorders* **29**(4), 275–305.
- Kloth, S., Kraaimaat, F., Janssen, P. and Brutten, G. (2000), 'Persistence and remission of incipient stuttering among high-risk children', *Journal of Fluency Disorders* **24**(4), 253–265.
- Kohonen, T. (1987), 'State of the art in neural computing'.
- Kramida, A., Yu. Ralchenko, Reader, J. and NIST ASD Team (2018), NIST Atomic Spectra Database (ver. 1.5), [Online]. Available: <http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sclite.htm> [2018, Jan 18]. National Institute of Standards and Technology, Gaithersburg, MD.
- Lamel, L., Gauvain, J.-L. and Adda, G. (2002), 'Lightly supervised and unsupervised acoustic model training', *Computer Speech & Language* **16**(1), 115–129.
- Langevin, M. (2009), 'The peer attitudes toward children who stutter scale: Reliability, known groups validity, and negativity of elementary school-age children's attitudes', *Journal of Fluency Disorders* **34**(2), 72–86.
- Langevin, M., Packman, A. and Onslow, M. (2010), 'Parent perceptions of the impact of stuttering on their preschoolers and themselves', *Journal of Communication Disorders* **43**(5), 407–423.

- Lass, N. J., Ruscello, D. M., Pannbacker, M. D., Schmitt, J. F. and Everly-Myers, D. S. (1989), 'Speech-language pathologists' perceptions of child and adult female and male stutterers', *Journal of Fluency Disorders* **14**(2), 127–134.
- Lee, S., Potamianos, A. and Narayanan, S. (1999), 'Acoustics of children's speech: Developmental changes of temporal and spectral parameters', *The Journal of the Acoustical Society of America* **105**(3), 1455–1468.
- Lever, J., Krzywinski, M. and Altman, N. (2016), 'Points of significance: Classification evaluation', *Nature Methods* **13**(8), 603–604.
- Liao, H., Pundak, G., Siohan, O., Carroll, M., Coccaro, N., Jiang, Q.-M., Sainath, T. N., Senior, A., Beaufays, F. and Bacchiani, M. (2015a), Large vocabulary automatic speech recognition for children, in 'Interspeech'.
- Liao, H., Pundak, G., Siohan, O., Carroll, M. K., Coccaro, N., Jiang, Q.-M., Sainath, T. N., Senior, A., Beaufays, F. and Bacchiani, M. (2015b), Large vocabulary automatic speech recognition for children, in 'Sixteenth Annual Conference of the International Speech Communication Association'.
- Liu, H., Gegov, A. and Cocea, M. (2016), Complexity control in rule based models for classification in machine learning context, in 'UK Workshop on Computational Intelligence', Vol. 513, Springer, pp. 125–143.
- Liu, Y., Shriberg, E., Stolcke, A. and Harper, M. P. (2005), Comparing hmm, maximum entropy, and conditional random fields for disfluency detection., in 'Interspeech', pp. 3313–3316.
- Lowerre, B. T. (1976), The harpy speech recognition system, Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE.
- Luo, G., Huang, X., Lin, C.-Y. and Nie, Z. (2015), Joint entity recognition and disambiguation, in 'Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing', pp. 879–888.

- Mahesha, P. and Vinod, D. (2012), 'Vector quantization and mfcc based classification of dysfluencies in stuttered speech', *Bonfring International Journal of Man Machine Interface* **2**(3), 1.
- Mahesha, P. and Vinod, D. (2015), Using orthographic transcripts for stuttering dysfluency recognition and severity estimation, in 'Intelligent Computing, Communication and Devices', Springer, pp. 613–621.
- Mahesha, P. and Vinod, D. (2016), 'Gaussian mixture model based classification of stuttering dysfluencies', *Journal of Intelligent Systems* **25**(3), 387–399.
- Martin, J. H. and Jurafsky, D. (2009), *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*, Pearson/Prentice Hall Upper Saddle River.
- Martinez, D., Plchot, O., Burget, L., Glembek, O. and Matějka, P. (2011), Language recognition in ivectors space, in 'Twelfth Annual Conference of the International Speech Communication Association'.
- Maskey, S., Zhou, B. and Gao, Y. (2006), A phrase-level machine translation approach for disfluency detection using weighted finite state transducers, in 'Ninth International Conference on Spoken Language Processing'.
- McCallum, A. and Li, W. (2003), Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons, in 'Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4', Association for Computational Linguistics, pp. 188–191.
- Middag, C., Saeys, Y. and Martens, J.-P. (2010), Towards an asr-free objective analysis of pathological speech, in 'Eleventh Annual Conference of the International Speech Communication Association'.
- Mooney, S. and Smith, P. K. (1995), 'Bullying and the child who stammers', *British Journal of Special Education* **22**(1), 24–27.

- Moore, J., Kronenthal, M. and Ashby, S. (2005), ‘Guidelines for ami speech transcriptions’, *AMI Deliverable* .
- Nöth, E., Niemann, H., Haderlein, T., Decher, M., Eysholdt, U., Rosanowski, F. and Wittenberg, T. (2000), Automatic stuttering recognition using hidden markov models, *in* ‘Sixth International Conference on Spoken Language Processing’ .
- Ntouro, K., Conture, E. G. and Lipsey, M. W. (2011), ‘Language abilities of children who stutter: A meta-analytical review’, *American Journal of Speech-Language Pathology* .
- Onslow, M. and Packman, A. (1999), *The handbook of early stuttering intervention*, Singular.
- Parveen, S. and Green, P. (2002), Speech recognition with missing data using recurrent neural nets, *in* ‘Advances in Neural Information Processing Systems’, pp. 1189–1195.
- Passos, A., Kumar, V. and McCallum, A. (2014), ‘Lexicon infused phrase embeddings for named entity resolution’, *arXiv preprint arXiv:1404.5367* .
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E. (2011), ‘Scikit-learn: Machine learning in Python’, *Journal of Machine Learning Research* **12**, 2825–2830.
- Pennington, J., Socher, R. and Manning, C. (2014), Glove: Global vectors for word representation, *in* ‘Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)’, pp. 1532–1543.
- Peters, T. J. and Guitar, B. (1991), *Stuttering: An integrated approach to its nature and treatment*, Williams & Wilkins.
- Peterson, L. E. (2009), ‘K-nearest neighbor’, *Scholarpedia* **4**(2), 1883.
- Pfau, T. and Ruske, G. (1998), Estimating the speaking rate by vowel detection, *in* ‘Tagungsband ICASSP 98’, pp. 945–948.

- Potamianos, A. and Narayanan, S. (2003), 'Robust recognition of children's speech', *IEEE Transactions on speech and audio processing* **11**(6), 603–616.
- Potamianos, Narayanan, L. (1997), Automatic speech recognition for children, in 'EUROSPEECH', EUROSPEECH, pp. 2371–2374.
- Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P. et al. (2011), The kaldi speech recognition toolkit, Technical report, IEEE Signal Processing Society.
- Powers, D. M. (2011), 'Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation'.
- Proença, J., Lopes, C., Tjalve, M., Stolcke, A., Candeias, S. and Perdigão, F. (2017), 'Detection of mispronunciations and disfluencies in children reading aloud', *Proc. Interspeech 2017* pp. 1437–1441.
- Rabiner, L. R. (1989), 'A tutorial on hidden markov models and selected applications in speech recognition', *Proceedings of the IEEE* **77**(2), 257–286.
- Rabiner, L. R., Juang, B.-H. and Rutledge, J. C. (1993), *Fundamentals of speech recognition*, Vol. 14, PTR Prentice Hall Englewood Cliffs.
- Randolph, J. J. (2005), 'Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa.', *Online submission* .
- Ratinov, L. and Roth, D. (2009), Design challenges and misconceptions in named entity recognition, in 'Proceedings of the Thirteenth Conference on Computational Natural Language Learning', Association for Computational Linguistics, pp. 147–155.
- Ratner, N. B., Rooney, B. and MacWhinney, B. (1996), 'Analysis of stuttering using chldes and clan', *Clinical linguistics & phonetics* **10**(3), 169–187.
- Ratner, N. B. and Tetnowski, J. A. (2014), *Current issues in stuttering research and practice*, Psychology Press.



- Ravikumar, K., Rajagopal, R. and Nagaraj, H. (2009), An approach for objective assessment of stuttered speech using mfcc, *in* 'The International Congress for global Science and Technology', p. 19.
- Reilly, S., Onslow, M., Packman, A., Wake, M., Bavin, E. L., Prior, M., Eadie, P., Cini, E., Bolzonello, C. and Ukoumunne, O. C. (2009), 'Predicting stuttering onset by the age of 3 years: A prospective, community cohort study', *Pediatrics* **123**(1), 270–277.
- Reynolds, D. A., Quatieri, T. F. and Dunn, R. B. (2000), 'Speaker verification using adapted gaussian mixture models', *Digital signal processing* **10**(1-3), 19–41.
- Richels, C. and Conture, E. (2010), 'Indirect treatment of childhood stuttering: Diagnostic predictors of treatment outcome', *Treatment of stuttering: Established and emerging interventions* pp. 18–55.
- Riley, G. (1994), *Stuttering severity instrument for children and adults*, Pro-ed.
- Robinson, A. (1996), 'The british english example pronunciation (beep) dictionary', *Retrieved from World Wide, Web: ftp://svrftp.eng.cam.ac.uk/pub/comp.speech/dictionaries/beep.tar.gz.*
- Robinson, T., Franssen, J., Pye, D., Foote, J. and Renals, S. (1995), Wsjcamo: a british english speech corpus for large vocabulary continuous speech recognition, *in* 'Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on', Vol. 1, IEEE, pp. 81–84.
- Russell, M. (2006), 'The pf-star british english children's speech corpus', *The Speech Ark Limited*.
- Russell, M. and D'Arcy, S. (2007), Challenges for computer recognition of children's speech, *in* 'Workshop on Speech and Language Technology in Education'.
- Seide, F., Li, G. and Yu, D. (2011), Conversational speech transcription using context-dependent deep neural networks, *in* 'Twelfth annual conference of the international speech communication association'.

- Sheehan, J. G. (1974), 'Stuttering behavior: A phonetic analysis', *Journal of Communication Disorders* **7**(3), 193–212.
- Shimada, M., Toyomura, A., Fujii, T. and Minami, T. (2018), 'Children who stutter at 3 years of age: A community-based study', *Journal of fluency disorders* **56**, 45–54.
- Snover, M., Dorr, B. and Schwartz, R. (2004), A lexically-driven algorithm for disfluency detection, in 'Proceedings of HLT-NAACL 2004: Short Papers', Association for Computational Linguistics, pp. 157–160.
- Stolcke, A. et al. (2002a), Srlm-an extensible language modeling toolkit., in 'Interspeech', Vol. 2002, p. 2002.
- Stolcke, A. et al. (2002b), Srlm-an extensible language modeling toolkit., in 'Interspeech', pp. 901–904.
- Sundaram, R. and Picone, J. (2004), Effects on transcription errors on supervised learning in speech recognition, in '2004 IEEE International Conference on Acoustics, Speech, and Signal Processing', Vol. 1, IEEE, pp. I–169.
- Suszyński, W., Kuniszyk-Józkowiak, W., Smółka, E. and Dzieńkowski, M. (2015), 'Prolongation detection with application of fuzzy logic', *Annales Universitatis Mariae Curie-Skłodowska, sectio AI-Informatica* **1**(1), 1–8.
- Sutton, S., Cole, R. A., Villiers, J. d., Schalkwyk, J., Vermeulen, P., Macon, M. W., Yan, Y., Kaiser, E., Rundle, B., Shobaki, K. et al. (1998), Universal speech tools: The cslu toolkit, in 'Fifth International Conference on Spoken Language Processing'.
- Świetlicka, I., Kuniszyk-Józkowiak, W. and Smółka, E. (2009), Artificial neural networks in the disabled speech analysis, in 'Computer Recognition Systems 3', Springer, pp. 347–354.
- Świetlicka, I., Kuniszyk-Józkowiak, W. and Smółka, E. (2013), 'Hierarchical ann system for stuttering identification', *Computer Speech & Language* **27**(1), 228–242.

- Szczurowska, I., Kuniszyk-Józkowiak, W. and Smółka, E. (2006), 'The application of kohonen and multilayer perceptron networks in the speech nonfluency analysis', *Archives of Acoustics* **31**(4 (S)), 205–210.
- Tan, T.-S., Ariff, A., Ting, C.-M., Salleh, S.-H. et al. (2007), Application of malay speech technology in malay speech therapy assistance tools, in 'Intelligent and Advanced Systems, 2007. ICIAS 2007. International Conference on', IEEE, pp. 330–334.
- Tran, Y., Blumgart, E. and Craig, A. (2011), 'Subjective distress associated with chronic stuttering', *Journal of fluency disorders* **36**(1), 17–26.
- Tseng, H., Chang, P., Andrew, G., Jurafsky, D. and Manning, C. (2005), A conditional random field word segmenter for sighthan bakeoff 2005, in 'Proceedings of the fourth SIGHAN workshop on Chinese language Processing', Vol. 171, Citeseer.
- Van Riper, C. (1973), *The treatment of stuttering*, Prentice Hall.
- Vertanen, K. (2007), 'Csr lm-1 language model training recipe'.
- Walden, T. A., Frankel, C. B., Buhr, A. P., Johnson, K. N., Conture, E. G. and Karrass, J. M. (2012), 'Dual diathesis-stressor model of emotional and linguistic contributions to developmental stuttering', *Journal of abnormal child psychology* **40**(4), 633–644.
- Ward, D. (2017), *Stuttering and cluttering: frameworks for understanding and treatment*, Psychology Press.
- Ward, L., Stefani, A., Smith, D., Duenser, A., Freyne, J., Dodd, B. and Morgan, A. (2016), Automated screening of speech development issues in children by identifying phonological error patterns, in '17th Annual Conference of the International Speech Communication Association (INTERSPEECH 2016)', pp. 2661–2665.
- Weninger, F., Geiger, J., Wöllmer, M., Schuller, B. and Rigoll, G. (2011), The munich 2011 chime challenge contribution: Nmf-blstm speech enhancement and recognition for reverberated multisource environments, in 'Machine Listening in Multisource Environments'.

- WHO (1957), 'Manual of the international statistical classification of diseases, injuries, and causes of death: based on the recommendations of the seventh revision conference, 1955, and adopted by the ninth world health assembly under the who nomenclature regulations'.
- Widrow, B. (1988), 'Darpa: Neural network study'.
- Wilpon, J. G. and Jacobsen, C. N. (1996), A study of speech recognition for children and the elderly, *in* 'icassp', IEEE, pp. 349–352.
- Wiśniewski, M., Kuniszyk-Józkowiak, W., Smółka, E. and Suszyński, W. (2007a), Automatic detection of disorders in a continuous speech with the hidden markov models approach, *in* 'Computer Recognition Systems 2', Springer, pp. 445–453.
- Wiśniewski, M., Kuniszyk-Józkowiak, W., Smółka, E. and Suszyński, W. (2007b), 'Automatic detection of prolonged fricative phonemes with the hidden markov models approach', *Journal of Medical Informatics & Technologies* **11**, 2007.
- Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., Yu, D. and Zweig, G. (2016), 'Achieving human parity in conversational speech recognition', *arXiv preprint arXiv:1610.05256*.
- Yairi, E. (1983), 'The onset of stuttering in two- and three-year-old children: A preliminary report', *Journal of Speech and Hearing Disorders* **48**(2), 171–177.
- Yairi, E. (2007), 'Subtyping stuttering i: A review', *Journal of fluency disorders* **32**(3), 165–196.
- Yairi, E. and Ambrose, N. (2013), 'Epidemiology of stuttering: 21st century advances', *Journal of fluency disorders* **38**(2), 66–87.
- Yairi, E. and Ambrose, N. G. (2005), *Early childhood stuttering for clinicians by clinicians*, Pro Ed.
- Yaruss, J. (1997), 'Clinical measurement of stuttering behaviors', *Contemporary Issues in Communication Science and Disorders* **24**(24), 33–44.

- Yaruss, J. S., Max, M. S., Newman, R. and Campbell, J. H. (1998), 'Comparing real-time and transcript-based techniques for measuring stuttering', *Journal of Fluency Disorders* **23**(2), 137–151.
- Yeh, P., Yang, S., Yang, C. and Shieh, M. (2015), 'Automatic recognition of repetitions in stuttered speech: Using end-point detection and dynamic time warping', *Procedia-Social and Behavioral Sciences* **193**, 356.
- Yu, D. and Seltzer, M. L. (2011), Improved bottleneck features using pretrained deep neural networks, in 'Twelfth annual conference of the international speech communication association'.
- Yuan, J. and Liberman, M. (2010), Robust speaking rate estimation using broad phonetic class recognition, in 'Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on', IEEE, pp. 4222–4225.
- Zebrowski, P. M. (1994), 'Duration of sound prolongation and sound/syllable repetition in children who stutter: Preliminary observations', *Journal of Speech, Language, and Hearing Research* **37**(2), 254–263.



# Appendix A

## Interview form

### A.0.1 Interview form : Questionnaire for parents

#### 1. Child's Personal Details

**ID:**

**Date of Birth:**

**Gender:**

- Male
- Female

How many People live at home?

How many children in the family?

#### 2. History of speech problem in the family

When did the child begin problem with his/her speech?

- Between 1 and 2 years.
- Between 2 and 3 years.
- Between 3 and 4 years.

- Between 4 and 5 years.
- Between 5 and 6 years.
- After 6 years.

Has one of the family members had stuttering in the past?

- Yes.
- No.
- I don't know.

Has this person had stuttering for a long time?

- Yes.
- No.
- I don't know.

Has this person recovered from stuttering?

- Yes.
- No.
- I don't know.



Which family member who suffer from stuttering?

- Mother.
- Father.
- Brother or sister.
- Uncle/ Aunt/ Cousin.
- I don't know.

### 3. Parental thoughts and feelings about your child's speech

I would rate my child's stuttering as:

1	2	3	4	5
Very mild				Very severe

How concerned are you about your child's speech?

1	2	3	4	5
Not worried				Very worried

Response to fluency:

1	2	3	4	5
Ignore the stuttering				Notify the child about the disfluency

How do the family try and help?

- Picking a simple words during the conversations.
- Ignore the stuttering when the child talk.



# Appendix B

## Reading passages

### B.1 ‘Arthur the rat’ passage by (Abercrombie, 1964)

There was once a young rat named Arthur who would never take the trouble to make up his mind. Whenever his friends asked him, if he would like to go out with them. He would only answer ‘I don’t know’ he would say yes and he wouldn’t say no either. He could never learn to make a choice. His aunt Helen said to him ‘no one ever care for you, if you carry on like this. You have no more mind than the blade of grass’. Arthur looked wise but said nothing. One rainy day the rats heard a great noise in the loft where they lived. The pine rafters were all rotten and the last one of the joists had given way and fallen to the ground. The walls shook and the rats’ hair stood on the end with fear and horror. ‘This won’t do’ said the old rat who was a who was chief. I’ll send out scouts to search for a new home. Three hours later the seven scouts came back and said ‘we found a stone house which is just what we wanted. There is room and good food for us all. There is a kindly horse named Nelly, a cow a calf and a garden with an elm tree’. Just then, the old rat caught sight of young Arthur ‘are you coming with us’ he asked. I don’t know Arthur sighed ‘the roof may not come down just yet’. ‘Well’ said the old rat angrily ‘we can’t wait all day for you to make up your mind’ right about face ‘march!’ and they went off. Arthur stood and watched the other rats hurry

off hurry away. The idea of intermediate decision was too much for him. 'I'll go back to my hole for a bit' he said to himself 'just make up my mind'. That night there was a great crash that shook the earth and down came the whole roof. Next day some men rode up and looked at the ruins. One of them moved a board and under it they saw a young rat lying on his side quite dead half in and half out of his hole

## **B.2 'One more Week to Easter' passage developed in UCL lab**

There is only one more week to Easter. I have already started my holiday. The idea of visiting my uncle during this Easter is wonderful. His farm is in this village down in Cornwall. This village is very peaceful and beautiful. I have asked my aunt if I can bring Sam, my dog, with me. I promise her I will keep him under control. He attacked and he ate some animals from her farm in October. But he is part of the family and I cannot leave him behind.

In my uncle's village, there is a shop which sells a lot of things. The owner of the shop has known my uncle since young. They were old friends. This shop sells kites among other things. I had been given one by my Dad. I couldn't believe my eyes when I opened it up. It was just an amazing thing. He had also taught me how to fly it. I enjoy it and it is fun. It brings me out to the open space more often. I am also seeing my Dad more often. He stays away from his office at weekends. That is good. There are all sorts of kites. I am actually getting myself some more during this visit.

During this visit, I'm going to ride a horse. I wasn't allowed to do that because I wasn't old enough. But I am older now. My aunt is very good at riding and she will teach me. I am asking to be allowed to stay with my aunt for the summer. I am excited by the idea. Besides riding, I can make some pocket money by helping out in the farm. They have got some cows and sheep, as well as pigs and some chickens. But I don't want to work with pigs.

They are about eight miles from the sea. My uncle can take me out on his new boat. He has spent his spare time building this boat. I am always being amazed by what he can do. He was an airman and his life is an interesting story on its own and he is full of surprises. In April, he is constructing a tree-house for Amy, my older cousin. Over Easter, it will be her eleventh birthday. If Mum lets me, I will buy her some chocolates. But Amy will be disappointed if I can't get her any. But I think she will understand. She has got no idea that I'm arranging a surprise party for her.

