TOWARDS A ROBUST ARABIC SPEECH RECOGNITION SYSTEM BASED ON RESERVOIR COMPUTING

ABDULRAHMAN ALALSHEKMUBARAK

Doctor of Philosophy

Computing Science and Mathematics

University of Stirling

November 2014

DECLARATION

I hereby declare that this thesis has been composed by myself and that it embodies the results of my own research. Where appropriate, I have acknowledged the nature and the extent of work carried out in collaboration with others included in the thesis.

Stirling, November 2014

Abdulrahman
Alalshekmubarak

ABSTRACT

In this thesis we investigate the potential of developing a speech recognition system based on a recently introduced artificial neural network (ANN) technique, namely Reservoir Computing (RC). This technique has, in theory, a higher capability for modelling dynamic behaviour compared to feed-forward ANNs due to the recurrent connections between the nodes in the reservoir layer, which serves as a memory. We conduct this study on the Arabic language, (one of the most spoken languages in the world and the official language in 26 countries), because there is a serious gap in the literature on speech recognition systems for Arabic, making the potential impact high. The investigation covers a variety of tasks, including the implementation of the first reservoir-based Arabic speech recognition system. In addition, a thorough evaluation of the developed system is conducted including several comparisons to other state-of-the-art models found in the literature, and baseline models. The impact of feature extraction methods are studied in this work, and a new biologically inspired feature extraction technique, namely the Auditory Nerve feature, is applied to the speech recognition domain. Comparing different feature extraction methods requires access to the original recorded sound, which is not possible in the only publicly accessible Arabic corpus. We have developed the largest public Arabic corpus for isolated words, which contains roughly 10,000 samples. Our investigation has led us to develop two novel approaches based on reservoir computing, ESNSVMs (Echo State Networks with Support Vector Machines) and ESNEKMs (Echo State Networks with Extreme Kernel Machines). These aim to improve the performance of the conventional RC approach by proposing different readout architectures. These two approaches have been compared to the conventional RC approach and other state-of-the-art systems. Finally, these developed approaches have been evaluated on the presence of different types and levels of noise to examine their resilience to noise, which is crucial for real world applications.

DEDICATION

To my parents

ACKNOWLEDGMENTS

I would like to express my deep gratitude to Professor Leslie S. Smith, my principal supervisor, for his support and guidance over the course of this PhD study. He has been always there for our weekly meetings despite his busy schedule. I have enjoyed our discussions during these meetings which were crucial to the success of this study. His supervision style, that provided me with the freedom to explore a variety of areas while ensuring that I was heading in the right direction, not only contributed to the completion of this work but also made it far more enjoyable. In addition, seeing his outstanding work ethic greatly inspired me to work hard even through the toughest times that typically every PhD students faces. I am honoured to finish my PhD study under his brilliant supervision. I would also like to thank Professor Bruce Graham for his valuable feedback and suggestions.

I am also grateful to the staff of King Faisal University for hosting my Arabic corpus project. They kindly provided me with everything I required to complete this project within the limited time line. Finally, I would like to thank my cousin, Dr. Ibraheem, for his support and wise words since my arrival in Scotland, seven years ago, to start my MSc course. His advice was very valuable and helped me during my stay in the UK while pursuing my higher education studies.

LIST OF PUBLICATIONS

During this PhD study, the following publications have been produced:

- Abdulrahman Alalshekmubarak and Leslie S. Smith. On Improving the Classification Capability of Reservoir Computing for Arabic Speech Recognition in Wermter, S., Weber, C., Duch, W., Honkela, T., Koprinkova-Hristova, P., Magg, S., Palm, G., Villa, A.E.P. (Eds.) , Artificial Neural Networks and Machine Learning-ICANN 2014, 24th International.

- Abdulrahman Alalshekmubarak and Leslie S. Smith. A noise robust Arabic speech recognition system based on the echo state network Acoustical Society of America 167th meeting, Providence RI, USA, 5-9 May 2014. (J. Acoustical Society of America, 135 (4) part 2, p2195).

- Abdulrahman Alalshekmubarak and Leslie S. Smith. A novel approach combining recurrent neural network and support vector machines for time series classification.Innovations in Information Technology (IIT), 2013 9th International Conference on, pp.42,47, 17-19 March 2013 doi: 10.1109/Innovations.2013.6544391

- Abdulrahman Alalshekmubarak, Amir Hussain, and Qiu-Feng Wang. Off-line handwritten Arabic word recognition using SVMs with normalized poly kernel. Neural Information Processing , pp. 85-91 Springer Berlin Heidelberg. 2012 the 19th International Conference on Neural Information Processing (ICONIP2012).

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AI	Artificial Intelligence
AN	Auditory Nerve
ANNs	Artificial Neural Networks
ASR	Automatic Speech Recognition
BPTT	Back-Propagation Through Time
DARPA	Defense Advanced Research Projects Agency
DNNs	Deep Neural Networks
DTW	Dynamic Time Warping
EKMs	Extreme Kernel Machines
ELMs	Extreme Learning Machines
ESN	Echo State Network
ESNEKMs	Echo State Network Extreme Kernel Machines
ESNSVMs	Echo State Network Support Vector Machines
FFT	Fast Fourier Transform
GMMs	Gaussian Mixture Models
GPGPUs	General-Purpose Computing on Graphics Processing Units
HMMs	Hidden Markov Models

IPA	International Phonetic Alphabet
LPC	Linear Predictive Coding
LS-SVMs	Least Squares Support Vector Machines
LSM	Liquid State Machine
LSTM	Long-Short-Term Memory
MFCCs	Mel-Frequency Cepstral Coefficients
ML	Machine Learning
MLPs	Multi-Layer Perceptrons
OAA	One Against All
OAO	One Against One
PGM	Probabilistic Graphical Model
PLP	Perceptual Linear Prediction
RASTA-PLP	RASTA-Perceptual Linear Prediction
RBF	Radial Basis Function
RC	Reservoir Computing
RNNs	Recurrent Neural Networks
SAD	Spoken Arabic Digits corpus
SLFNs	Single-hidden Layer Feedforward Networks
SVMs	Support Vector Machines
VC	Vapnik-Chervonenkis theory

1

INTRODUCTION

Since the rise of the digital world, developing a machine that can perform cognitive tasks such as speech recognition has been a major aim in academia and in the commercial sector. The significant potential of such a machine was recognised by all of the contributing parties in the early days of the computational era. This resulted in the emergence of this new discipline that aims to pursue this objective, namely Artificial Intelligence (AI). AI can be seen as the field that borrows all of the needed concepts and techniques from many different domains, such as linguistics, neuroscience, statistics, etc. and utilises them to create an intelligent machine. Despite the efforts and resources that have been made during the past few decades, this task has proven very challenging and the development of such a machine has started to be considered by the general public as science fiction. This difficult beginning for the field led to a decline in the interest of the community and many scholars shifted away from this field, as did the available funding during the 1980s and 90s. This, however, has started to change as AI began to make major progress in recent years due to the recent development of machine learning (ML) which is nowadays widely considered to be the most active branch of AI.

These recent advances in the AI field and the big data era that the world is experiencing (where the amount of digital data is exponentially increasing) have contributed to the rise of the field. This means that there is an even greater need for a machine that can mine and take advantage of these huge resources. In addition, government agencies are no longer the main funding resources for research projects in the field and very influential bodies such

as the Defense Advanced Research Projects Agency (DARPA) are challenged by relatively young commercial players such as Google and Microsoft. This involvement by the private sector is due to the significant economical value of the research conducted in the field. There is an unprecedented race in the commercial world to adopt and develop ML techniques to gain a competitive edge in the market and to exploit the benefit of mining the petabytes of data on their private servers and all over the internet. This race can be clearly seen from the recent recruitment of main figures in the ML field by several leading companies, such as Geoffrey Hinton by Google in 2013, Andrew Ng by Baidu in 2014 and Yann LeCun by Facebook in 2013.

Speech recognition systems, which fall under the natural language processing umbrella, in particular has witnessed a major breakthrough based on the advances in ML in recent years. The success in developing new techniques to construct and train artificial neural networks (ANNs), the feed forward deep learning paradigm, has resulted in the wide adoption of the ANNs model in speech recognition applications. The acoustic modelling phase, which is a crucial component of the state-of-the-art speech recognition architecture, is now completely dominated by the ANNs approach. The commercial world has also witnessed the wide adoption of this technology and many of the major commercial speech recognition systems have announced the use of the feed forward deep learning method in their application. This includes the well-known Android operating system developed by Google, and Microsoft has also announced that it is planning to adopt this technology to provide online speech translation through one of its main applications, namely Skype. To sum up, the advances in the feed forward deep learning paradigm have resulted in significant excitement in the field, and many long-promised systems by the AI community that have the potential for overcoming the language differences in speech communication are finally materialising.

In this PhD study, we built on this recent success in the ANNs domain and investigated the potential of developing a speech recognition system based on another recently introduced ANNs technique, namely reservoir computing (RC). This technique has, in theory, far more capability in terms of modelling dynamic behaviour compared to the feed forward deep learning approach, due to the recurrent connections among the nodes in the reservoir layer which serve as a memory to the system. We conduct this study on the Arabic language which is one of the major spoken languages in the world and the official language in 26 countries. This selection of Arabic is based upon identifying a serious gap in the literature compared to other languages and the potential impact of improving speech recognition systems for such a widespread language.

This investigation covers a variety of tasks, including the implementation of the first reservoir-based Arabic speech recognition system. A thorough evaluation of the developed system is conducted and several comparisons are made between it, the state-of-the-art models found in the literature and the baseline models. The impact of the feature extraction methods are also studied in this work and new feature extraction techniques, the AN-based feature and RASTA-PLP, are applied for the first time to the Arabic domain. Comparing different feature extractions methods requires access to the raw recordings files which is impossible in the only publicly accessible Arabic corpus so during this PhD we developed the largest public Arabic Corpus for Isolated Words that contains roughly 10,000 samples and 50 participants. Our investigation has led us to develop two novel approaches based on reservoir computing (ESNSVMs and ESNEKMs), that aims to improve the performance of the conventional RC approach by proposing different system architectures. These two novel approaches have been compared to the conventional RC approach and other state-of-the-art systems not only from the performance perspective but also from others perspectives, such as complexity, stability,

training difficulties, etc. Finally, the resilience of these implemented systems under noise is also covered in this work and the developed systems have been evaluated in the presence of different types and levels of noise.

1.1 MOTIVATIONS

Several factors motivated us to select the Arabic speech recognition and reservoir computing domains as the topic for this PhD study. The potential impact of this work in academia and the commercial world is one of the major motives in conducting this study. This includes the fact that many applications rely heavily on speech recognition systems which means that the improvement in this domain will have a significant impact on a broader range of domains. Real time translation and human computer interfaces are good examples of such domains, where the speech recognition system is the main component. In such domains, improving speech recognition systems is crucial for improving the overall performance of the considered application.

The recent advances in the ANNs domains and in RC in particular are another main motive behind the selection of this area of research. The advances in training recurrent neural networks, enabled by the introduction of RC, allows researchers to construct and train large networks, that contain over a million nodes, effectively using the computational power available on medium stream, off-the-shelf computers. This means that, unlike other conventional recurrent training methods, where typically the error is propagated through the entire network, such as the short-long term memory approach, there is no need to use specialised hardware or collect a very large corpus to train the model. RC also offers a significant reduction in the training time. This attractive property is associated with models that use random projection, such as the extreme learning machines technique. This reduction in time allows scholars to experiment with a variety of novel designs in a short time, which is

crucial to the success of projects that need to be conducted within a relatively limited time.

In addition, the limited work conducted on Arabic does not reflect its importance in the world. This gap in the literature played a major role in the selection of this topic, as we believe that far more efforts need to be directed to the Arabic domain not only by academia but also by the private sector as well. Scholars who attempt to work in the Arabic domain are faced with serious challenges, such as the limited public resources and limited events, conferences and workshops devoted to this field. This results in a very poor publication presence in the literature compared to other languages and reduces the quality of the conducted research due to consuming a significant share of the research in preparing a small, in-house corpus to employ to evaluate the developed system. Finally, there was also a personal motivation that stems from the desire to contribute something back to society through improving the Arabic speech recognition domain.

1.2 OBJECTIVES

Developing an ambitious yet realistic research objective is crucial to ensure success when conducting PhD studies. A well-designed objective provides clear, important guidance to scholars throughout the various stages of the study. In addition, it not unusual in PhD studies, and in research in general, that choices need to be made at different stages and specifying the research objectives typically present researchers with convenient criteria that help them to make such selections. Thus, in this work we have focused on honing the research objectives and have revisited them during the course of this study.

The main objective of this study is to investigates the potential of applying the recently developed reservoir computing technique for an Arabic speech

recognition system. There are many sub-objectives that branch from this main research objective:

a) To implement a reservoir computing-based Arabic speech recognition system *b*) To evaluate the performance of the developed system *c*) To investigate the impact of the feature extraction methods on system performance *d*) To investigate the impact of the activation functions on system performance *e*) To develop a novel-based system based on reservoir computing *f*) To evaluate the developed system in the presence of noise.

These stated objectives have shaped the work conducted in this PhD study. This includes the design and implementation of a variety of experiments and their following evaluation and comparison. In the conclusion chapter we will evaluate our success in achieving these research objectives and state the implications of our findings.

1.3 RESEARCH CONTRIBUTIONS

In this section, we state the original contributions to the field achieved by this PhD study, starting from the earliest to the latest, adopting date-based sorting criteria:

- To develop the first reservoir-based speech recognition system and compare it with other state-of-the-art published work using the well- known SAD corpus (this work has been published)[2].
- To develop a novel reservoir computing-based speech recognition system that combines ESN and SVMs and compare it with other state-of-the-art published work and the conventional ESN using the well-known SAD corpus (this work has been published)[2].

- To develop the largest publicly accessible corpus for Arabic isolated words that contains about 10000 audio files (samples) uttered by 50 speakers (this work is under preparation for publication).

- To develop a novel reservoir computing-based Arabic speech recognition system that combines ESN and EKMs and compares to ESN, ESNSVMs, state-of-the-art published work and HMMs base line models (this work has been published)[3].

- To apply a novel feature extraction technique, AN-based feature, and conduct empirical comparisons between different state-of-the-art feature extraction techniques under different acoustic environments (this work has been published)[3] [4].

- To develop a noise robust Arabic speech recognition system architecture that applies RASTA-PLP in the feature extraction process and our developed approach ESNEKMs on the classification stage (this work has been published)[3].

These continuations are described and discussed over the course of this thesis and the implications of our work are stated in the conclusion chapter.

1.4 STRUCTURE OF THESIS

The remainder of this thesis is organised as follows:

- Chapter two introduces the field of automated speech recognition systems and establishes the concepts and terminology related to this work. All of the applied feature extraction methods are also covered in this chapter.

- Chapter three gives a brief introduction of the machine learning domain and all the related classification approaches that are adopted or discussed in this work.

- In Chapter four, we present the first reservoir-based Arabic speech recognition system, and evaluate it on three different corpora. These three corpora are also described, including our self-developed corpus.

- The two novel reservoir-based approaches (ESNSVMs and ESNEKMs) are presented and evaluated in Chapter five. The effect of noise on system performance are also discussed in this chapter. Finally, the impact of the activation function on system performance is also included in this chapter.

- We discuss the findings of Chapters four and five in Chapter Six and highlight promising area for future work.

- Finally, we conclude in Chapter seven by a presenting a brief summery and revisit our research objectives to draw our conclusion.

2

AUTOMATED SPEECH RECOGNITION

2.1 INTRODUCTION

Speech communication is one of the most distinguishing abilities of humans. In fact, in the well-known Turing test, the ability to conduct a conversation was introduced in the early days of computation as a measurement of the intelligence level. Automatic speech recognition (ASR), mapping an acoustic signal into a string of words, forms the first part of this intelligent system. ASR has been applied successfully in a wide range of real-world applications with different levels of success; however, the design of a robust ASR system is still an open challenge.

The first ASR system can be traced back to the 1950s when the first spoken digits recognition system for a single speaker was introduced[23]. Different ASR systems were also developed at the same time and focused on phoneme recognition (mostly vowel detection systems). These models were very limited due to the computational power constraints at that time and the absence of the required knowledge to build such systems. These first attempts reveal the need to develop more representative features for the acoustic signal and an effective pattern recognition algorithm. Linear Predictive Coding (LPC) was introduced to overcome the shortage of the feature extraction methods, whereas Dynamic Time Warping (DTW) was adopted to classify the patterns of different audio signals and also for the variation in the speech of the utterances.

A variety of models have been proposed over the years but without much success. The major breakthrough was the introduction of the hidden Markov

model (HMMs)[10]. HMMs quickly came to dominate the field due to its robust performance in isolated and continuous speech recognition systems[81]. This, however, has started to change as artificial neural network (ANN) have shown outstanding performance in many tasks, such as image and speech recognition. ANNs have been applied since the 1980s, but the lack of efficient learning algorithms has limited their adoption in the field[68]. The main reason behind their recent success is the development of new learning techniques that handle a complex network topology, such as deep feedforward networks and large size recurrent networks[21][49]. Reservoir computing is one of the recently developed approaches to train recurrent networks, and it has proven to be very successful in many applications (see chapter 4 for a formal introduction).

2.2 STATE-OF-THE-ART ARCHITECTURE OF IN AUTOMATED SPEECH RECOGNITION

Over the past 60 years a significant body of knowledge has been developed that focuses on designing the most effective architecture for ASR systems. The development of such an architecture needs to be based on the ASR problem, which can be described as follows. Given a set of observations O ,where $O = o_1, o_2, \ldots, o_t$ extracted from the acoustic signal we would like to predict the corresponding set of words string W, where $W = w_1, w_2, \ldots, w_n$. In the language L the objective is [61]:

$$\hat{W} = \arg\max_{W \in L} P(W|O)$$

and by using Bayes' rule we can rewrite the previous expression as :

$$\hat{W} = \arg\max_{W \in L} \frac{P(O|W)P(W)}{P(O)}$$

since $P(O)$ is the same for all the sentences in the language, we can further simplify the equation:

$$\hat{W} = \arg\max_{W \in L} \frac{P(O|W)P(W)}{P(O)} = \arg\max_{W \in L} P(O|W)P(W)$$

where $P(O|W)$ is known as the acoustic model and P(W) is the language model. O is commonly obtained by dividing the signal in the time domain into overlapping segments (known as frames, commonly 25ms and 10ms for the size and the shift of the frame). These extracted frames are processed by a feature extraction method, e.g. MFCCs. The previously described steps are known as front-end processing and the resulting vectors are used to train the acoustic model. In the unsupervised mode, a clustering algorithm is applied, such as k-means, and Baum-Welch [81] is used to train HMMs. The target labels and the processed frames are directly fed to a discriminative classifier (MLPs, SVMs) to learn the acoustic model in the supervised mode. The language model is learnt as an N-gram language model in which its order depends on the vocabulary size and available data for training. A dynamic programming algorithm (namely, Viterbi) is used to obtain the most probable sequence of words, giving the acoustic model and the language model. This is known as the decoding process.

This model is discussed throughout this thesis, and a formal introduction is presented for each of its components. The front-end process is described later in this chapter, whereas the training of the language model, acoustic model and the decoding process are included in Chapter three.

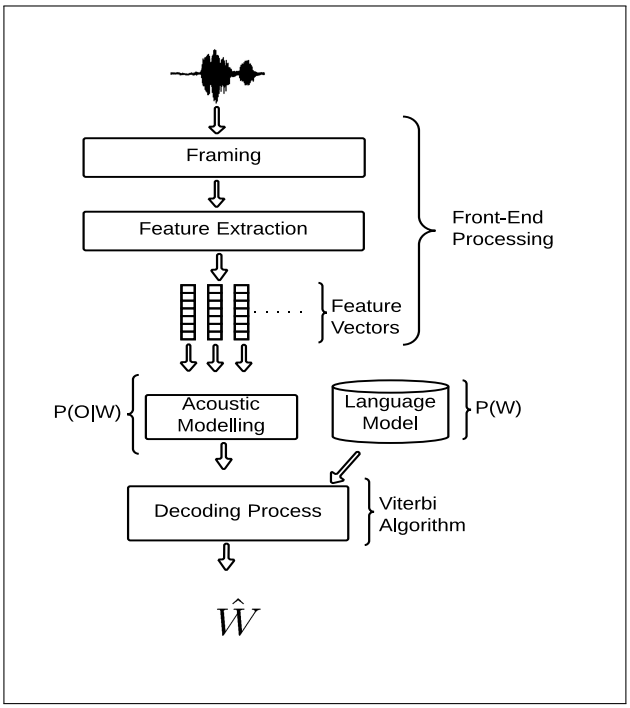

Figure 2.1: A block diagram of the state-of-the-art architecture of ASR.

2.3 THE ROLE OF LINGUISTICS IN AUTOMATED SPEECH RECOGNITION

Linguistics, which is the study of human language [46], is one of the major fields that has contributed to the design of modern ASR systems. Historically, linguists tried to develop ASR based purely on the grammar of the language and to build a rule-based representation that governs the behaviour of the language. However, designing such a system has proven to be impractical (not only in the ASR but also across all language based application, e.g. speech tagging), this due to the fact that speakers do not always use grammar. Statistics, on the other hand, provides a superior performance in real-world applications with regard to the cost of making assumptions, mostly inconsistent with the rules developed by linguists, to simplify the task of modelling the human language. This has led to a significant shift in the community. The problem of designing ASR is largely considered to be the statisticians' job, and the knowledge offered by linguists is regarded as irrelevant and should not influence

the design of ASR systems. This started to change especially in the front-end phase when feature extraction methods based on phoneme analysis and the modelling of the human auditory system with a high level of abstraction, such as MFCCs, out-performed the state-of-the-art technologies at the time.

Recently, ideas from the fields of linguistics and neuroscience have been adopted in the front-end processing stage and classification stages. In addition, many scholars have suggested a more balanced approach that combines knowledge from different fields, including linguistics, to overcome the weaknesses of the current ASR systems. This direction is encouraged by today's increasing computational power that allows researchers to investigate new models that, until recently, were considered intractable systems. The lack of novel advances in the statistical approach during the past decade and the fact that almost all of the increases in the performance were achieved by adopting hybrid approaches (such as ANN-HMMs) also encourage this new trend.

2.3.1 *Human Speech*

Today, the mechanism for producing or receiving speech in humans is fairly well understood as a physical phenomenon[99]. Only the transformation from thoughts to neural impulses and the interpretation of the preprocessed signal, i.e., the recognition of a word based on the spikes received by the brain, remain unknown. In other words, the cognition segment of the process is still missing from the big picture, but a detailed description is available for the speech production system and the preprocessing steps in the auditory system that transfers the sound wave into electric pulses (action potential). Human speech is in the range of 80 Hz to 7 kHz, and humans hear sounds that fall between 20 Hz and 20 kHz, though this range tends to decrease with age. These numbers make it more plausible to discard all of the frequencies that are not in the human speech range while building ASR systems to avoid the interference

of any background noise. In addition, the logarithmic behaviour found in the human cochlea has promoted the use of Mel scales and other scales that have proven to be very effective. Recently, novel feature extraction methods with a relatively low level of abstraction have been proposed to convert the acoustic signal into spike trains and have shown promising results, especially in noisy environments[72]. In addition, the analysis of spoken languages is also important in designing feature extraction methods. In particular, the analysis of phonemes is very useful in building a system that can correctly classify a word by identifying the phonemes that of which it is composed. These phonemes are not consistent across languages or, indeed, accents in the same languages , promoting a multidisciplinary approach to designing phoneme-based extraction features[100].

2.3.2 *Phonetics & Phonology*

Phonetics, which can be defined as ẗhe study of the physical aspects of the speech events, including speech production, speech acoustic and speech perception, and phonology, which is the study of the interaction between the sound segments of a language (also known as phones), contributes significantly to the modern ASR [46].This is particularly true in continuous speech ASR; however, they influence all types of ASR systems. Phonetics focuses on studying phones, which are the basic units of sound that a language contains. There are differences between using phones in different languages as some languages, such as Arabic, involve one-to-one mapping between the written system and the phonetic sound. This mapping is not present in many languages, such as English and French. The idea of decomposing speech into its basic sound segments and extracting the grammar that governs their interaction was introduced by Noam Chomsky and Morris Halle in 1968 in their well-known book " The Sound Pattern of English "[19]. However, the ASR mod-

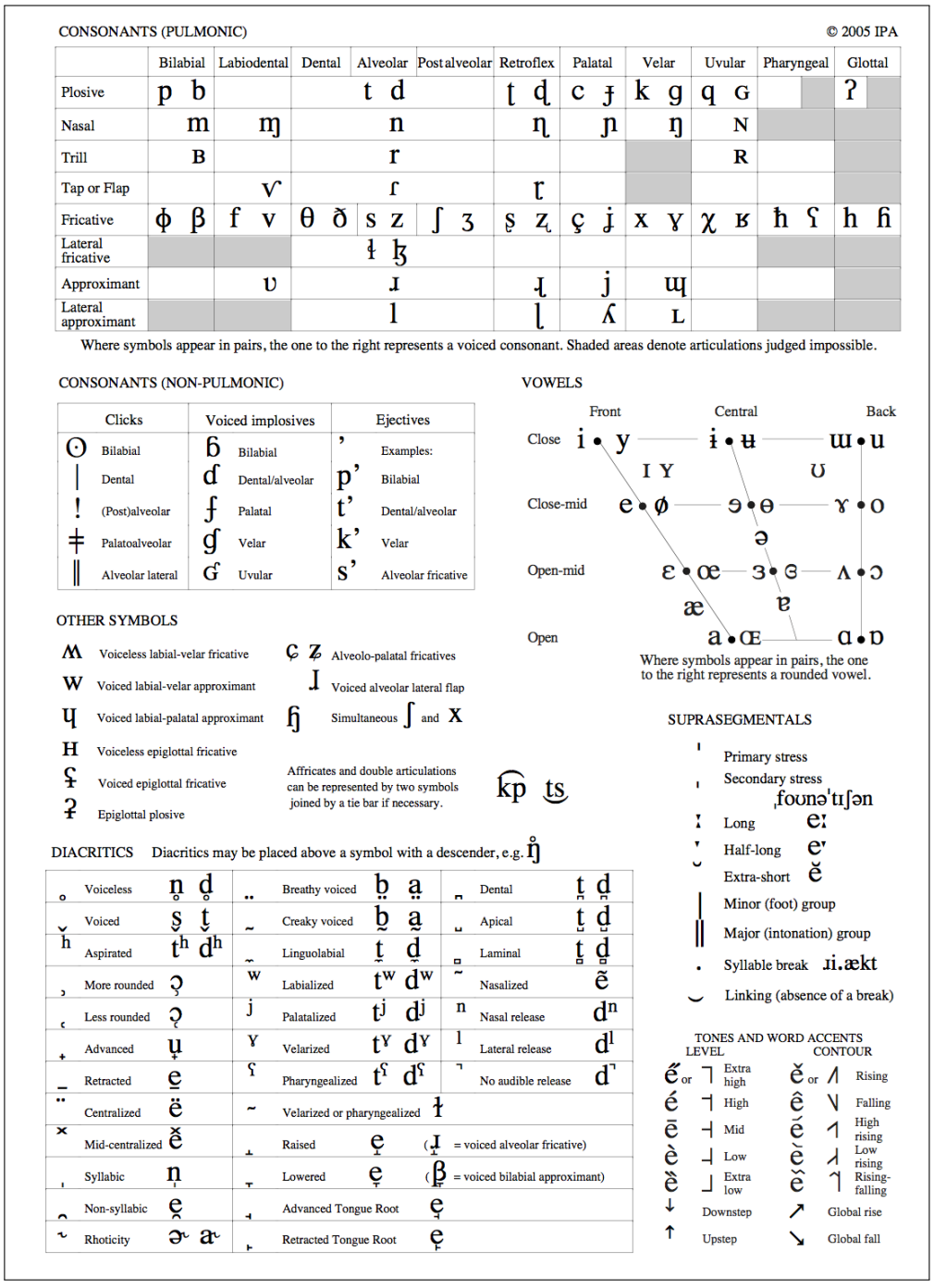

Figure 2.2: The International Phonetic Alphabet (revised to 2005) chart adapted from [64].

els at the time were not adequately advanced to make use of this knowledge. The first successful attempt to integrate this knowledge in developing ASR models could be traced back to the mid-1970s when a small number of groups

(Bell Labs, Carnegie-Mellon and IBM [33]) demonstrated the capability of HMMs for designing such systems in small vocabulary tasks. It took until the mid-1980s to develop ASR for a large vocabulary in which HMMs started to dominate the field and the computation power became available to implement such systems.

The International Phonetic Alphabet (IPA) is a worldwide standard for all of the phones present in all languages (see Figure 2.2). Phones are divided into two parts: vowels, phones produced with an open vocal tract, and consonants, phones produced with a partially closed vocal tract. Consonants are classified by the place of articulation and the manner of their articulation. One of the major insights that phonetics offers the domain of ASR is that the number of phones is restricted by the physical constraints across languages, and many of these phones are shared between different languages. This concept led to the suggestion that combining resources across languages in learning the acoustic model could be beneficial. Though this argument had been proposed earlier in the field, it has only been recently demonstrated to be practical [25]. Phones are useful in detecting different languages, accents and dialects.

In addition, phonology provides crucial information about how phones interact, which is more language-specific. This information is commonly integrated in the modern continuous ASR architecture in the language model. Another major contribution is the articulation theory, which states that the pronunciation of phones is affected by the preceding and following phones. Currently, acoustic modelling approaches rely heavily on this theory as it has proved to have a significant impact on performance, and today, it is common practise to build systems with the triphone classification model to account for these phenomena. This means that, for a language such as English, the number of different classes for a phone-based system will increase dramatically from 44 to 85184 in a triphone-based system, more generally from N phones classes to N^3 [65]. Despite this increase in model complexity, this approach has been

shown to be very effective but requires very significant amounts of training[65]. The number of phones varies among the languages' accents and this needs to be addressed in the designing phase. In other words, phonetics-based ASR systems allow researchers to take advantage of the similarity in the pronunciation of different languages but this implies that these systems cannot directly cope with large changes in pronouncing of words that appear in different accents within the same language.

In conclusion, knowledge developed in phonetics and phonology is very important in state-of-the-art ASR systems. Many theories proposed in both fields have been considered to be the common practice in designing modern ASR systems. Based on the success of these systems, a more integrated (multidisciplinary) approach that combines knowledge from different disciplines has been suggested and promoted recently in the literature [100].

2.4 FEATURE EXTRACTION APPROACHES

In the feature extraction process, the aim is to represent the acoustic signal in a compressed format that maintains the necessary information to perform the required task (in this case, ASR). Ideally, this representation will be invariant to changes due to noise and auditory environments or the characteristics of the speaker while being sensitive to change due to pronouncing different utterances. It was clear in the early days of ASR that representing the signal in the time domain does not achieve the desired properties as the acoustic signal varies significantly even if the same utterances are made by the same speaker[61]. This discovery led the community to search for alternatives to the time domain representation, and to the adoption of the frequency domain representation as it has proven to be a more robust approach that also models the auditory preprocessing mechanism to a relatively high level of abstraction. All of the state-of-the-art feature extraction methods involve converting the

signal from the time domain into the frequency domain where spectral analysis is performed.

Many of the spectral analysis methods have been inspired by the human auditory system. This can be seen in the development of different critical band filter scales based on the crude approximation to the behaviour of the cochlea, but the most widely-adopted scales are Mel and Bark (see Table 2.1). The community quickly realised the importance of this approach as this rough modelling of the human auditory system provides robust representations. Most of the state-of-the-art methods are currently using these scales (e.g., the Mel scale used in MFCCs and the Bark scale used in PLP). However, despite this early success in modelling auditory systems, developing more biologically realistic models has proven difficult, and no significant advances have been made in pursuing this approach during the past two decades. The last widely-adopted method was PLP-RASTA, which was developed in 1997. This is mainly due to the fact that the classifiers used in the acoustic modelling stage require low dimensional representations in order to provide a satisfactory performance. The Gaussian Mixture Model (GMM) is the most dominant method that requires small and uncorrelated features. This shows the issues raised in considering each stage separately and ignoring the interaction among the tasks. A clear example can be seen when a self-taught representation has been suggested in[82], where a more realistic classification method is proposed based on deep neural networks with a very basic preprocessing step that outperforms MFCCs and PLP-based systems. In other words, the success of developing more biologically plausible feature extraction method depends on the ability of the classification method used in the acoustic modelling phase, and there is evidence in the literature to suggest that adopting biologically inspired approaches in the feature extraction and acoustic modelling phases improves ASR performance.

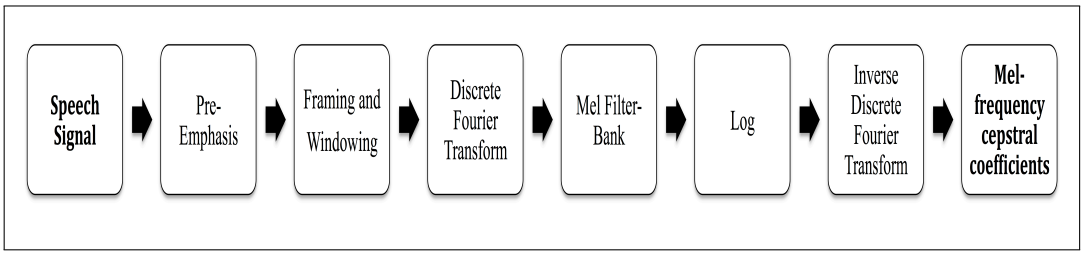

Figure 2.3: A block diagram of MFCC generation.

In this study, a more realistic approximation of the human auditory system suggested in[87] is adopted and compared with other state-of-the-art methods, namely, MFCCs, PLP and RASTA-PLP. The selection of this method is based on its recent success in the onset classification tasks[78]. This adoption is possible as we use the reservoir computing technology in the classification stages, allowing us to use a large number of features while maintaining a reasonable learning time.

In conclusion, designing a robust representation has proven to be a very challenging task and is still an open area for research[8]. The main approach in developing feature extraction techniques is to model the human auditory system or certain aspects of it. A variety of techniques have been proposed with different levels of abstraction. The reminder of this section discusses the different approaches considered in this study.

2.4.1 *Mel-frequency Cepstral Coefficients*

In ASR systems, using MFCCs is by far the most widely-adopted approach. The process of computing MFCCs from the acoustic signal consists of six steps, which are shown in Figure 2.3. The first step is pre-emphasis, which increases the energy in the high frequencies, and then the signal is divided into frames using an overlap moving window, with a frame size of 25 ms. A 10 ms shift size and Hamming window are the standard parameters used in this step.

Bark Scale		Mel Scale	
Centre Frequency	Critical Bandwidths	Centre Frequency	Critical Bandwidths
50	100	100	100
150	100	200	100
250	100	300	100
350	100	400	100
450	110	500	100
570	120	600	100
700	140	700	100
840	150	800	100
1000	160	900	100
1170	190	1000	124
1370	210	1149	160
1600	240	1320	184
1850	280	1516	211
2150	320	1741	242
2500	380	2000	278
2900	450	2297	320
3400	550	2639	367
4000	700	3031	422
4800	900	3482	484
5800	1100	4000	556
7000	1300	4595	639
8500	1800	5287	734
10500	2500	6063	843
13500	3500	6964	969

Table 2.1: Bark and Mel filter bank scales adapted from [62].

The discrete Fourier transform is applied to the extracted frames commonly computed by the fast Fourier transform (FFT) for efficiency. The resulting

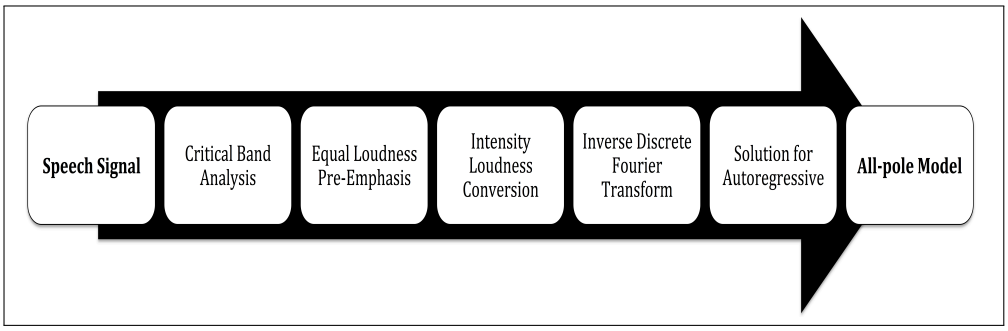

Figure 2.4: A block diagram shows the steps of computing PLP.

values are then mapped by the Mel filter bank, which has 1000 Hz threshold value so that all of the values below it map linearly and all the values above it map logarithmically, using the following equation[61] :

$$Mel(f) = 1127 \ln(1 + \frac{f}{700})$$

Finally, the inverse discrete Fourier transform is calculated, resulting in the final components of the vector. This vector is then used in the classification stage. The typical size of this vector is 13, 12 cepstral coefficients combined with the frame energy.

2.4.2 *Perceptual Linear Prediction*

Perceptual Linear Prediction was proposed in[47] as a technique that is more consistent with human hearing, and has been successfully applied to a variety of systems. Figure 2.4 shows a block diagram of this model. The main strength of this technique is the ability to compress speaker-dependent information while maintaining the relevant information needed to identify different linguistic traits even when a small number of orders is used. This property, a low-dimensional representation of the signal, is considered very useful in the classification stage as many classification techniques tend to provide a higher performance under such regimes. The main limitation of this approach,

however, is its sensitivity to noise, which can limit its adoption in real-world applications.

2.4.3 *RASTA- Perceptual Linear Prediction*

In order to overcome the limitations of PLP, the RASTA-Perceptual Linear Prediction (RASTA-PLP) approach has been developed[48]. It provides a low-dimensional representation with robust performance in noisy environments. Unlike short-term spectral analysis, RASTA-PLP makes use of the context information. In other words, RASTA-PLP could be seen as an attempt to shift the focus of the field from frame-by-frame analysis toward context analysis, which is believed to be more consistent with the human auditory system. In addition, RASTA-PLP has been proven to be successful for other tasks, such as speech enhancement.

2.4.4 *Auditory Nerve Based Feature*

Auditory nerve (AN) is a biologically-inspired approach that models the behaviour of the mammalian auditory nerve. This method was introduced a decade ago, but it has not been implemented in speech recognition tasks[87]. Recently, an onset classification system that combines the AN feature with the echo state network was introduced in[78], and it has been proven to be very effective. One of the main attractions of this method is that, unlike the previously described method, the signal is analysed in the time domain instead of the spectral domain, allowing for more precise time-event detection.

2.4.4.1 *Auditory Nerve Based Feature's Computed Process*

The steps of computing the auditory nerve-based feature is shown in the figure 2.5. The AN-based feature consists of different levels in a hierarchical fashion, inspired by the hierarchical nature observed in the biological system. Level 0 is the first level. At this level the acoustic signal is recorded and converted to the digital form. Once the analogue signal has been transformed to the required form, it is passed through a cochlea-like filter, namely, the gammatone filter. The number of the used bandpass filters is a system-dependent parameter; in this work we found 64 bands sufficient for the task of designing Arabic speech recognition systems. The output of each band is then used to generate the spike-based representation. Different spike trains are computed for each channel, allowing the approach to cover a wider range. A single spike is generated in a positive-going zero crossing mode in a way that allows the same spike to be recorded in a different spike trains. In other words, if a spike is generated in spike train s, then all spike trains s' that fall in range $0 < s' < s$ will record this spike as well. Finally, the different spikes trains of each band are combined to produce the level 0 AN-based feature[87].

The level 1 AN-based feature consists of two different approaches. The first method applies the Gabor filter on the level 0 AN-based feature, and a single or multiple Gabor filters can be used here. The use of the Gabor filter has been encouraged by evidence found in the literature that claims that biological data can be modelled using the Gabor filter [73]. The second method focuses on utilising the onset detection system to compute a robust representation for ASR systems. This onset detection system takes the level 0 feature as an input and passes it through a depressing synapse to an onset neuron, which is a leaky integrate-and-fire cell. The leakage level of the onset neuron controls the sensitivity of the system to detect the onset. In other words, selecting a high level of leakiness prevents the onset neuron from firing, meaning that it

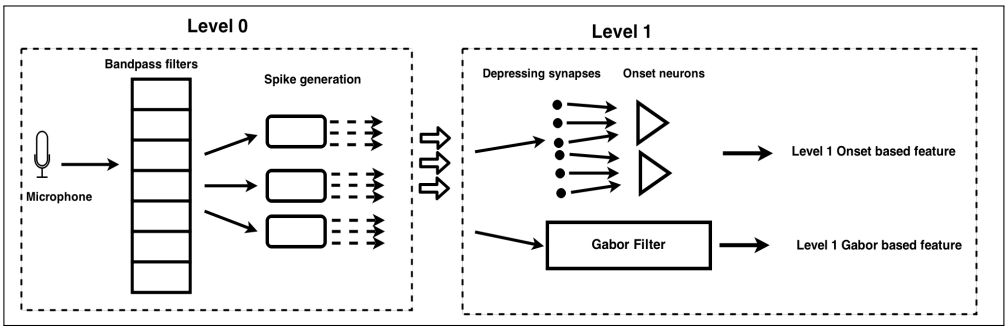

Figure 2.5: A block diagram shows the steps of computing AN based feature.

misses the onset events. The outputs of these cells are combined to form the level 1 onset-based feature [86].

To the best of our knowledge, this hierarchical AN-based feature has not been applied to speech recognition. The main barrier to a wider adoption of this method is that the dimension of the resulting feature is high compared to that of standard methods, such as MFCCs or PLP. This forms a challenge for the standard classification methods, e.g., HMMs. In ESN, this problem is less serious as it can efficiently handle high-dimension regimes due to the random initialisation. See section 3.3.3.5 for a detailed description. In chapter five, we report the results obtained by this approach and compare them with those produced by the standard methods (MFCC, PLP and PLP-RASTA).

3

MACHINE LEARNING FOR AUTOMATED SPEECH RECOGNITION

It is essential to introduce the field of machine learning (ML) in discussing ASR systems as many ML techniques are used in the acoustic modelling and the language modelling stages. The difficulties found in the early days of ASR, during which the focus was primarily on handcrafting rules that were designed by experts, encouraged the shift toward the ML approach. These difficulties stem from the challenges related to extracting such rules, which also tend to be language dependent. This limits knowledge sharing across language domains and requires considerable labour.

In addition, designing these rules requires a solid understanding of human cognitive perception, which is still largely unknown. The ML concept offers a different perspective for tackling these issues. ML suggests that, instead of focusing on developing these rules manually, we should focus on designing techniques that can *learn* these rules. The learning process involves presenting these algorithms as a set of examples in datasets. Although many of these extracted rules are domain specific and language dependent, the ML techniques are universal and can be applied across languages by using different datasets.

In other words, ML provides the means by which unknown functions can be approximated based on collected examples. It is important to state that ML techniques have achieved significant success across different fields and that they are considered to be state-of-the-art techniques in many fields related to cognitive behaviour, such as computer vision and language processing. In this chapter, we introduce ML formally and emphasise the relationship between

the two fields. The widely adopted techniques are presented, together with a brief history of their introduction into ASR. The assumptions that are being made, as well as the weaknesses and strengths of the considered algorithms, are also discussed.

3.1 MACHINE LEARNING

Machine learning is defined as *"computational methods using experience to improve performance or to make accurate predictions"* [77]. From this definition, it is clear that datasets and algorithms play an important part in developing successful ML applications. During the past two decades, ML has witnessed a rapid increase in its popularity as the available computation power has increased, novel algorithms have been proposed and more datasets have been made available. ML is successfully applied in a variety of fields, such as medicine, finance and artificial intelligence.

The main concepts of ML is to give the machine the ability to learn from the data to perform tasks without being explicitly programmed. There are many different perspectives for interpreting the goal of ML. A major perspective is to consider ML as an optimisation problem wherein the aim is to minimise the mismatch error between the output hypothesis and the desired output. Another perspective that is common in the literature and is influenced by statistics is to regard the ML problem as a probability density function estimation problem. Many ML techniques have been developed and introduced based on these two points of view.

ML can also be seen as an induction problem wherein the objective is to generalise based on a finite example. This sheds the light on the importance of generalisation in ML, which means that the computed hypothesis can cope with unseen examples. Inappropriate generalisation, commonly known as high variance or overfitting, leads to poor performance on unseen examples

while the model maintains a high level of accuracy on seen data. The Vapnik-Chervonenkis theory (VC) provides a regress mathematical model that links the generalisation performance with the number of samples. This theory is used to obtain an upper bound for generalisation errors in supervised learning. The relationship between the generalisation errors and the number of samples is expressed mathematically as follows[1]:

$$N \geq \frac{8}{\epsilon^2} \ln \left(\frac{(4(2N)^{d_{vc}} + 1)}{\delta} \right) \tag{3.1}$$

where N is the training example,ϵ is the error tolerance , δ the probability ϵ is violated and d_{vc} is the VC dimension. From the previous equation it can be calculated that for $\epsilon = 0.1$ and $\delta = 0.1$ almost 10,000 samples need to be added for every increase in the VC dimension by 1 to maintain the same bound. This is very difficult to implement in practice; thus, a value of 10 is used (which is much lower) instead as a rule of thumb to maintain the same bound, or more commonly, the cross-validation approach is used to estimate the generalisation performance. There is a growing amount of literature in the field that addresses the problem of generalisation, and different approaches, which tend to be algorithm specific, have been developed. These methods are included in the discussion of the considered classification techniques later in this chapter. A related problem to overfitting is underfitting, or high bias, which describes the phenomenon whereby the accuracy of the proposed model is poor on both seen and unseen data.

In developing ML applications, it is essential to recall that "all models are wrong but some are useful" [14]. In addition, the Occams's razor principle, which states that simpler models should be preferred, governs the development process in ML tasks. This means that the objective of ML is to construct the least complex model that can perform the task at hand with the required

accuracy. The complexity of a model is commonly measured by its number of parameters.

3.2 TYPES OF LEARNING IN MACHINE LEARNING

Machine learning contains three different main subfields that are categorised based on the learning type: supervised learning, reinforcement learning and unsupervised learning. Although all of these categories learn from the data, the formalisation of the problem is different in each type. In this section, we formally introduce these types and discuss the similarities and differences between them.

3.2.1 *Supervised Learning*

Supervised learning is the most widely discussed type in the literature. The dataset in supervised learning contains the input signal and the teaching signal (the desired outputs). The objective in the learning settings is to find the function that maps the input to the target output. Classification and regression tasks are commonly formulated as supervised learning problems, introduced later in this section. A typical example of supervised learning is the problem of face detection in which the aim is to classify images into two groups; one contains faces, and the other does not. The datasets in this example are a set of images and a label target for each image. The previous classification example is a relatively easy problem as it includes only two different classes. This is known as a binary classification, whereas a task that includes more classes, such as the ImageNet classification task, in which the dataset includes 1,000 classes, is more challenging. The evaluation in this type is based on approximating the mapping function. The main work of this study

is considered with regard to supervised learning, although many suggested concepts can be easily extended to other types.

3.2.2 *Unsupervised Learning*

In unsupervised learning, the dataset does not include a target. The objective here is to discover novel representations and relationships in the data. Clustering, which is the task of dividing a dataset into groups of classes, is the most widely-utilised type[7]. This, however, has recently started to change as unsupervised learning has proven to be very effective as a preprocessing step in training deep neural networks[49] (see section 3.3.1.5 for more details).

The evaluation criteria are unclear, as there is no agreed method for assessing the discovered knowledge. Thus, it is common to evaluate unsupervised methods in classification tasks. This evaluation method has been criticised in the literature due to the fact that evaluating the discovered knowledge in the targeted labels does not consider other novel representations that can prove more useful. A good example of this issue can be seen in applying a clustering algorithm to data that contain the age, gender and income of a group of people and then evaluating the output in the classification task that divides the data point based on income. This constrains the discovered knowledge to a single hypothesis and diminishes the initial objective of the unsupervised learning approach in discovering novel representations as the clustering algorithm can group the data based on other properties, such as age and, while this model gives poor performance on selected classification tasks, it offers a novel insight into data that can also be useful for other classification tasks. In other words, it is difficult to evaluate unsupervised learning specifically on the clustering approach as the aim is to discover novel representations, and there is no standardised method for choosing one representation over others, at least in the general context of unsupervised learning. In ASR, clustering is

used in the acoustic modelling phase to assign phoneme membership to the feature vectors. These membership values can be discrete as in the k-means algorithm or continuous, commonly interpreted as probability, as in gaussian distribution. A version of unsupervised learning is also used in training HMMs and depends on maximising the overall model probability if the used corpus does not contain phones-level labels (see section 6.1 for more details).

3.2.3 *Reinforcement Learning*

Reinforcement learning can be seen as an attempt to model the trial-and- error learning behaviour observed in the animal world. The data in this type do not include the desired output explicitly; instead, a reward function is incorporated into the model. This reward function grades the outputs of the model, and the objective of the reinforcement learning algorithm is to maximise the rewards. However, what makes this learning paradigm more challenging is that the rewards or penalties are not typically supplied after each action. This means that the model needs to memorise the performed actions and identify and select those actions that maximise the rewards. This type of learning is mostly adopted in agent-interaction settings and goal-driven tasks. A typical example of a reinforcement learning problem is the following: suppose we have agent A in position x, and the task is to move A to a new position x_{new}. The data in this example are a set of positions, each associated with actions, and the reward function is used on arrival at x_{new}. Although this is a very limited example that fails to present the strong aspects of the learning approach, it does capture the main differences between its problem settings and the other two learning approaches. Unlike supervised and unsupervised learning, reinforcement learning is not used in the state-of-the-art ASR systems; thus, this method is not discussed further in this work (for a detailed review, refer to [11]).

Classification is a major subfield of machine learning that has enjoyed rapid growth in the last decade. The classification problem can be summarised as follows:

Given a dataset D ,which is sampled from a subspace S, that contains n examples of the form $< \vec{x}, y >$where $\vec{x}$ is known as the feature vector (or the input vector) and y is the class target - $y \in \{0, 1\}$in the case of binary problems and $y \in \{0, 1, ..(m-1)\}$ where m is the number of classes in the case of multi label problems. We would like accurately to assign a new feature vector $\vec{x}_{new}$ $(\notin D)$to its class. In other words, we assume that there is an unknown function $f(\vec{x})$ that maps the input vector to a class label and the aim of a learner is to approximate this function.

The previous brief description shows the broad applicability of classification techniques in different, not necessarily related, areas, explaining its wide adoption in a variety of fields, e.g., medical diagnoses, natural language processing, computer vision, financial market prediction, security, etc. Despite the success in the field in recent years, there is still an ongoing need to develop a more efficient and robust classification algorithm that can open up the doors to further adoption.

A crucial concept in the classification world is the *no-free-lunch theorem*, which informally states that there is no single learning algorithm that is superior across all tasks[98]. In other words, for the classification problem, there is no one-size-fits-all solution that can be applied without considering the precise nature of the task. A good example can be seen when we consider the state-of-the-art approaches in tasks that are fundamentally different, such as text classification and computer vision applications. In text classification, there is a need to deal with regimes in which the input vector is extremely long and very sparse, and high-dimensional SVMs and Naive Bayes learners

are considered to be the state-of-the-art approaches[93]. However, in the computer vision application, the state-of-the-art approach is the neural network, particularly the convolutional neural network.

This increases the complexity of developing classification-based applications, as a sound knowledge of learning approaches combined with familiarity with the task at hand is required to design an effective system. This suggests conducting not only theoretical studies but also empirical ones as both types are essential to identify the strengths and weaknesses of individual learners. Researchers in the field have recognised the potential benefits of combining different classification algorithms to improve performance, giving rise to an active research area known as an ensemble technique[26][13]. However, developing an ensemble algorithm is a very challenging task as a sound theoretical understanding of the combined algorithm and excellent implementation skills are required[63]. In addition, conducting intensive experiments is critical to ensuring that valid conclusions are reached.

3.3.1 *Static Machine Learning Algorithms*

In this section, we introduce the static algorithms that are adopted to develop or compare models in this thesis. The discussed algorithms can be categorised based on the decision boundary as linear classifiers and nonlinear classifiers. Linear regression, the Widrow-Hoff rule and logistic regression fall into the linear classification category, whereas Multi-Layer Perceptron, support vector machines, least squares support vector machines, extreme learning machines and extreme kernel machines are nonlinear classifiers. All of these considered approaches are known as discriminative classification wherein the goal is directly to estimate the probability of the different classes, given input, $P(Y|X)$. Generative classifiers transform the former quantity using Bayes' theorem to $P(X|Y) * P(Y)$, which is easier to compute in most cases. The main reason

behind choosing not to adopt the generative approach in this work is that generative classifiers tend to require a regime in which the number of features is relatively small unless the inputs are discrete, and these features are not correlated in order to provide competitive performance. This regime is difficult to obtain in the reservoir computing context where the number of features tends to be very large with a high degree of correlation between features due to the random mapping used in the reservoir. In the next section, the considered algorithms are discussed, starting with the linear classification approaches and then moving to the nonlinear classifiers.

3.3.1.1 *Linear Regression*

Linear regression has a long history in the world of statistics; however, here we restrict the discussion to the use of linear regression in the context of classification problems. Although linear regression is considered to be limited due to its linear decision boundary, it is still applied in many applications today. This is mainly due to several attractive properties, including convex optimisation (single solution), fast convergence and low complexity, which are reflected in a good generalisation performance. Formally, linear regression can be introduced as follows: Given a dataset D that contains a set of examples $\vec{x}^i \in X$ and the targeted labels $\vec{y}$ where $y^i \in \{0,1\}$, for binary tasks, the objective is to minimise the following cost function:

$$\frac{1}{N}\sum_{i=1}^{N}(h(\vec{x}^i) - y^i)^2 \tag{3.2}$$

where N is the number of sample and h is defined as:

$$h(\vec{x}) = \vec{w}^\top \vec{x} \tag{3.3}$$

where $\vec{w}$, the weights vector, can be obtained by the following analytic formula:

$$\vec{w} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \vec{y} \tag{3.4}$$

From the previous discussion, we can see that this algorithm provides a single-shot solution, which is considered to be one of its main attractions. In order to overcome the linear decision boundary, X is commonly mapped into a higher dimension space where linear separation among classes is possible. In a regime in which the number of features is large relative to the sample size, regularisation is crucial. The regularised linear regression, also known as ridge regression, can also be obtained from an analytic formula as follows:

$$\vec{w} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \vec{y} \tag{3.5}$$

where $\mathbf{I}$ is the identity matrix and λ is the regularisation parameter, which is typically set by a grid search using the validation set. Linear regression is sensitive to the outlier presented in the data, which is one of its main limitations. Another major limitation of linear regression is that its output can be negative, which makes it difficult to interpret it as probability.

3.3.1.2 *The Widrow-Hoff Rule*

The Widrow-Hoff rule, also known as the delta rule, is one of the most widely-adopted training algorithms for single-layer neural networks. It is very similar to the linear regression algorithm if a linear activation function is used. In addition, it is also a linear classifier with a convex cost function which means that the optimisation surface does not contain local optima solutions. However, unlike linear regression, the solution is obtained via an iterative procedure wherein the weights matrix is initialised to random values, typically zero. This

weight vector is updated in each iteration until convergence is achieved. The delta rule can be expressed mathematically as follows:

$$\min E = \frac{1}{N}\sum_{i=1}^{N}(h(\vec{x^i}) - y^i)^2$$

where N is the number of sample and h is the computed using the formula in (3.3) if a linear function is applied. The weight vector is updated as follows:

$$\vec{w}_j := \vec{w}_j - \alpha\frac{1}{N}\sum_{i=1}^{N}(h(\vec{x^i}) - y^i)x_j^i \qquad (\textit{ simultaneously update } w_j \textit{ for all } j) \quad (3.6)$$

where y is the target output, α is the learning rate and N is the number of sample. Backpropagation, the most widely-applied learning algorithm in MLPs, is considered to be a generalisation of the delta rule (see section 5 for more details). In the case of using the linear activation function, linear regression provides a faster convergence and fewer hyperparameters, e.g., optimising the learning rate. However, in a regime in which both the number of data samples and the number of features are very large, the delta rule can be implemented in an online or mini-batch fashion. In other words, the delta rule offers two major advantages over linear regression. First, it generalises over different activation functions, which are the basis of MLPs. Secondly, an online or mini-batch version can be adopted to scale up the algorithm to encompass very large data sets.

3.3.1.3 *Logistic Regression*

Logistic regression is one of the most widely-used algorithms in machine learning. The high level of transparency, the ease of implementation and the ability to scale to very big data sets in the online or mini-batch learning mode are among its main attractions. For these reasons, in certain fields, such as medical machine learning applications, in which a high level of transparency is required, logistic regression is considered one of the state-of-the-art algorithms.

Unlike linear regression, only classification problems can be handled in logistic regression. The sigmoid function is used as an activation function, which has a bounded output value between 0 and 1. Mathematically, logistic regression can be expressed as follows:

$$h(\vec{x}) = g(\vec{w}^\top \vec{x})$$

where

$$g(z) = \frac{1}{1 + e^{-z}}$$

and the optimisation cost function is :

$$\frac{1}{m} \sum_{i=1}^{m} [-y^{(i)} \log(h(\vec{x}^{(i)})) - (1 - y^{(i)}) \log(1 - (h(\vec{x}^{(i)}))]$$

The optimisation cost function shown above is convex, which is another major attraction for logistic regression as all of the convergence issues related to local optima solutions are avoided. Multi-class classification tasks can be handled effectively with logistic regression. Typically, a one-versus-all approach is adopted, making it possible to scale it up to encompass a relatively large number of classes.

3.3.1.4 *Perceptron*

The perceptron algorithm was developed under the umbrella of computational neural network research (also known as artificial neural network, cognitive computing) and can be traced back to 1943 when Warren McCulloch and Walter Pitts published their influential paper entitled "A Logical Calculus of Ideas Immanent in Nervous Activity"[74]. The perceptron was introduced to

the field in 1957 by Frank Rosenblatt in [84].The objective of this field is to develop an artificial neural system that can be embedded in a machine to allow it to perform cognitive tasks. This machine is expected to capture a form of the intelligence found in humans. It is important to state that the development of such a machine promotes many challenging philosophical and psychological questions that are related to intelligence, consciousness and self-consciousness. These concepts will not be further discussed as they fall beyond the scope of this study and are not typically considered within the machine learning community (refer to [67] and[34]for additional discussions).

It is clear from the previous introduction that the computational neural network research is heavily related to the study of neural systems. In other words, the suggested abstracted artificial model reflects to a large extent our understanding of neural systems. The perceptron algorithm is not an exception as it is a crude model of the neuron, which is a specialised nerve cell that transmits electronic signals known as spikes. The perceptron enjoyed a period of popularity in the mid-1960s, but this popularity diminished in the late 1960s when Minsky and Papert proved that a single-layer perceptron cannot handle non-linearly separable systems[76].

Formally, the perceptron classifier attempts to model a simple neuron by a thresholded linear function. This function combines the input signal linearly and generates a binary output (fire or not) based on a learnt threshold. The function can be mathematically described as follows:

$$h(\vec{x}) = \text{sign}(\vec{w}^\top \vec{x})$$

where x is the input signal and w is the weights (learned parameters) that are calculated by an iterative procedure, known as the perceptron learning rule, as follows:

$$\vec{w}_{(t+1)} = \vec{w}_{(t)} + \alpha(y - \text{sign}(\vec{w}_{(t)}^{\top}\vec{x}))\vec{x}$$

where y is the target output, α is the learning rate and t indicates the number of iteration. This learning rule is guaranteed to converge for linearly separable data in a finite number of iterations. The main weakness of the perceptron learning rule is it that cannot handle nonlinearly separable data, which limits its adoption in real-world applications.

3.3.1.5 *Multi-Layer Perceptron (MLP)*

Multi-Layer Perceptrons (MLPs) were introduced in the 1980s as a solution to the limited classification capability of the single perceptron. The main concept here is that, by combining a collection of perceptrons and constructing a layered network of perceptrons, a nonlinearly separable function can be learnt. This concept had been known for a long time before the development of the Multi-Layer Perceptron algorithm as the initial aim of the perceptron was to model the human brain, which is estimated to have approximately 100 billion neurons. However, the issue of constructing a network of perceptrons was the lack of a practical learning algorithm. In the MLPs technique, the learning problem is solved by replacing the hard threshold function with a differentiable function to allow the use of a generalised version of the delta rule, known as the backpropagation algorithm[85]. It is important to note that MLP is a feedforward structure, which means it cannot handle temporal data.

The development of the MLPs technique started a new era of artificial neural network popularity. This era ended in the mid 1990s due to several issues related to the backpropagation algorithm. A major issue was that the training algorithm cost function is not a convex surface, meaning that the algorithm can be trapped in a local minimum solution. This also makes the algorithm very sensitive to the initial values. Another major issue was the computation cost

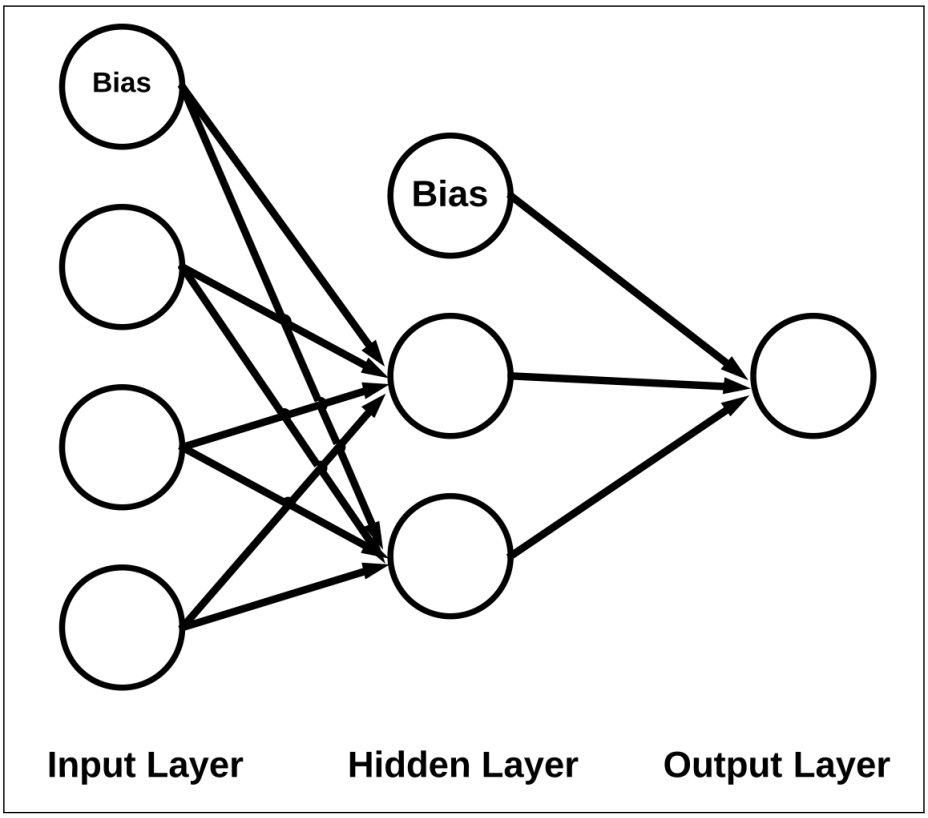

Figure 3.1: A simple example of Multi-Layer Perceptron to demonstrate its basic structure and the its different layers.

as training a network for real-world tasks requires a large number of nodes wherein a large dataset is required. In addition, overfitting is a serious issue in a large network, and a significant level of expertise is essential to handle such a network. The algorithm is also very sensitive to several hyperparameters, such as the learning rate, which tends to be very difficult to optimise.

Formally, MLP consists of neurons that are organised in a layered fashion. The first layer is known as the input layer, whereas the final layer is called the output layer. All of the layers that are not input or output layers are known as hidden layers. The network is considered deep if it has more than a single hidden layer. The input is passed through the network layer by layer, and finally, the value of the output layer is compared to the target label to obtain the error. This computed error is then used to backpropagate, hence the name backpropagation, through the network to adjust the weights.

Recently, a particular structure of the MLPs, namely, the deep neural network(DNNs), has shown significant performance mainly due to two main reasons. The first factor of the resurgence of the MLP is the increase in the computational power, which is achieved by the development of the general-purpose computing on graphics processing units (GPGPUs) approach in the mid-2000s[15]. The second reason is the increase of the data volume available to train such deep architectures. Despite this new success, there is a major limitation of the MLP technique, even the deep structure, that makes it impractical for many real-world tasks as it is fundamentally unable to handle dynamic systems directly.

Hyper-parameters Optimisation

One major problem in developing the MLP classifier is optimising the model's hyperparameters. These parameters include the number of layers and the number of nodes in each layer, known as the network topology. There is not an agreed approach to select the topology of the network, and commonly, this is performed in a problem-specific fashion. A major factor that needs to be considered in optimising the network structure is the size of the dataset (generally, the larger the dataset, the larger the network that is used, assuming enough computation power). In addition, the type of function used in each nodes is also an important choice which needs to be addressed in constructing MLP model. Typically, logistics or tanh are used, and recently rectifier nonlinearities have shown superior performance[71]. The main constraint on selecting the nonlinear function is to have an easily computed derivative.

The initial values of the weights are also important parameters that need to be optimised. Typically, the network's weights are randomly initialised, or a heuristic rule is adopted, such as the one introduced in [36]. The learning rate, which for normalised inputs commonly takes a value between 1 and 10^{-6} , is optimised in a grid search mode, and a default value of 0.01 has

been suggested in the literature[12]. In summary, the common approach to optimise the MLP model is to perform a grid search to adopt the default values suggested in the literature.

As we have discussed earlier, MLP has the tendency to overfit the training data. Thus, in optimising the model's hyperparameters, an emphasis should be made on controlling the model complexity to avoid this problem. Over the past decade, many techniques have been developed to tackle this problem. Penalising the network's weights is one of the main approaches in which the cost function is changed to include the weights of network. The complexity is controlled by λ, which is the hyperparameter to control the trade-off between fitting, classification of the error, and minimising the weights. In the weight decay method, typically, the squared sum of the weights is minimised, known as the L2 norm. This is equivalent to enforcing a gaussian prior with zero mean over the weights. Soft-weight sharing is also one of the developed approaches to prevent overfitting, and it was introduced in the early 1990s [79]. This method groups the weight in different clusters using gaussian models. Another regularisation technique is early stopping, in which a held-out subset from the training set is used to detect overfitting during the learning process. The training is terminated once the performance in this held-out subset starts to decrease. A more recently introduced regularisation technique is the drop-out approach, which aims to limit co-adaptation between units. This is being achieved by randomly switching off (dropping out) units during the training phase. Although this is a relatively new technique, its popularity has been growing rapidly due to its superior performance in different tasks[88].

3.3.1.6 *Support Vector Machines*

SVMs are state-of-the-art algorithms that were developed by Vapnik in the 1990s[93]. The first version of SVMs, namely, hard margin, was very limited as it could handle only linearly separable data, which prevents implementing

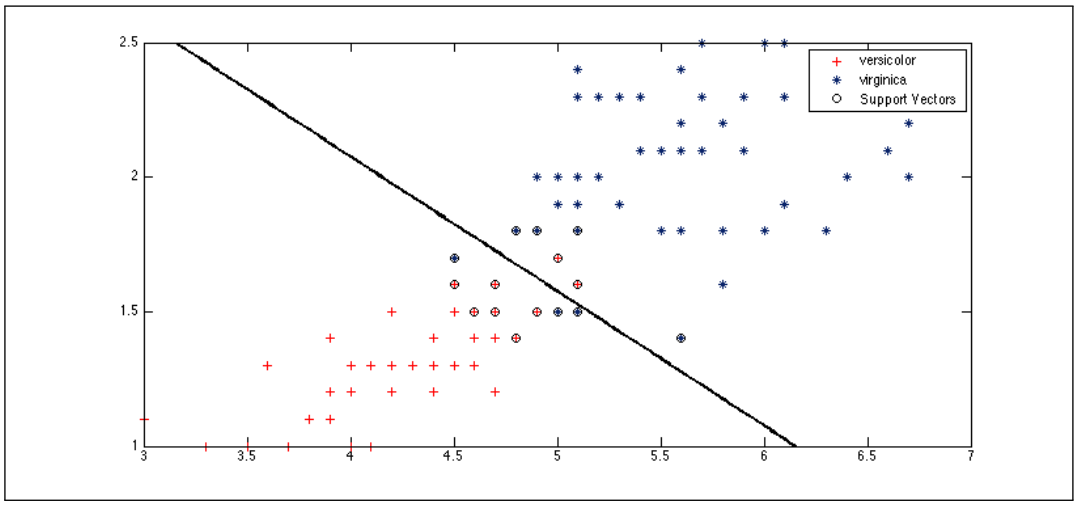

Figure 3.2: Illustration of the Decision Boundary of Linear SVMs.

SVMs in real-world tasks. The rise of SVMs started when the soft margin version, which has the ability to deal with more challenging tasks and noisy data in which data points are not linearly separable, was introduced in the mid-1990s[22]. It is important to state that SVMs can be seen as a weighted-instance-based algorithm that selects data points (support vectors) by assigning to them non-zero weights. The solution of SVMs is typically very sparse, meaning that in ideal conditions only a very small fraction of the training sample will be chosen as support vectors.

Many properties of SVMs have contributed to their popularity, including the robust performance, fast training, reproducible results and the sparse solution. The error bound of SVMs, which is critical for real-world applications, is also relatively easy to compute. The success of SVMs has drawn much attention in the field in the past two decades, and many models have been proposed under the field of kernel methods.

The main concept of SVMs can be informally described as follows. SVMs map the input vector $\vec{x}$ to a higher dimensional feature space using a kernel,

which is any function that satisfies the Mercer's condition[1], and finds the optimal solution that achieves the maximum margin. There are a variety of kernels that are typically applied in SVMs, such as linear, polynomial and radial basis functions (RBF). Developing new kernels for specific tasks is also an active area of research. The selection of a kernel affects the performance of the model, and commonly, the kernel is chosen and its parameters optimised using a cross-validation set. The main equation of SVMs that is used to estimate the decision function from a training dataset is stated as follows [93]:

$$h(x) = \text{sign}\left(\sum_{n=1}^{l} y_n \alpha_n . k(x, x_n) + b\right) \tag{3.7}$$

where l is the number of support vectors, b is the bias term, $y_n \in \{-1, +1\}$ is the class sign to which the support vector belongs and α is obtained as the solution of the following quadratic optimisation problem:

$$\begin{aligned} &\min \tfrac{1}{2} w^T w + C \sum_{i=1}^{p} \xi_i \\ &\text{s.t } y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i \\ &\xi_i \geq 0, i = 1, ...p \end{aligned} \tag{3.8}$$

The major limitation of SVMs is their inability to handle dynamic systems. This limitation can be addressed by converting time series to fixed long vectors before applying SVMs. However, this approach can make the resulting vectors very long, resulting in the curse of dimensionality, severely affecting

1 $K(x, x')$ is a valid kernel iff [22]:
1. K is symmetric.
2. K is positive semi-definite for any $x, \ldots, x_n$.

performance. The binary nature of SVMs can also be considered one of their limitations. In order to overcome this limitation and to extend SVMs to multi-label tasks, different techniques are used. The main approaches are 'one against all' (OAA) and 'one against one' (OAO), in which in the first approach N SVM classifiers will be built, one for each class, $\frac{(N-1)N}{2}$ and binary SVMs classifiers are implemented with the second approach [75]. The majority voting among these classifiers will be used in predicting new points. However, OAA and OAO are computationally expensive and can only be feasible when the number of labels is relatively small[52].

3.3.1.7 *Least Squares Support Vector Machines*

Least squares support vector machines (LS-SVMs) were introduced in 1999 by Suykens and Vandewalle[89]. LS-SVMs replace the inequality constraints of SVMs with equality constraints, meaning that the optimisation problem can be solved using linear programming instead of quadratic programming. However, the number of support vectors is now proportional to the number of errors. The sparse solution of SVMs is not maintained in LS-SVMs, which means that the number of support vectors in SVMs is typically smaller than that in LS-SVMs. The solution can be obtained by solving a linear system [57] which means that the implementation of LS-SVMs is much easier than implementation of SVMs. This and the superior generalisation performance obtained by LS-SVMs over SVMs are considered its main attractions. However, LS-SVMs still rely on a binary technique and cannot handle dynamic systems, which is the same shortcoming of SVMs.

In summary, LS-SVMs aim to reduce the complexity of the training phase of SVMs by replacing the inequality constraints with equality constraints. This allows the use of linear programming instead of quadratic programming, which decreases the complexity of the algorithm implementation. LS-SVMs

can generalise better than SVMs but the sparse solution of SVMs is no longer maintained.

3.3.1.8 *Extreme Learning Machines*

The extreme learning machines (ELMs) is proposed as an efficient model to train the single-hidden layer feedforward networks (SLFNs), which was developed by Huang in 2004 [54]. The basic concept is similar to reservoir computing in that both approaches map the input to a higher dimensional space using random weights and only learn the weights of the output layer. The main difference is that ELM, unlike RC, does not use recurrent nodes, which prevents it from modelling dynamic systems. ELM has been applied successfully in many real-world conditions in a variety of fields [55] [56].

Instead of using a relatively small number of nodes in the hidden layer and applying a powerful optimisation technique such as the back-propagation algorithm, which suffers from several well-known issues (e.g. local minima, very sensitive to initialisation weights, implementation complexity, tendency to over-fit, long training time), ELM uses a very large number of nodes, typically more than 1,000 and only uses a simple read-out function at the output layer. Despite the use of this large number of nodes, ELM offers superior generalisation performance, which can be explained by the random weights applied on the learning and testing samples. This means the mapping mechanism is not based on the learning dataset. ELM can be described mathematically as follows [55] :

$$f(\vec{x}) = \sum_{n=1}^{L} \beta_i h_i(\vec{x}) = \mathbf{h}(\vec{x})\vec{\beta} \tag{3.9}$$

where $\vec{\beta} = [\beta_1, ..., \beta_L]^T$ is the output learnt weights by the simple linear read-out function, L is the number of nodes and $\mathbf{h}(\mathbf{x}) = [h(x)_1, ..., h(x)_L]^T$ is calculated by mapping the input vector by the random initialised weights.

A variety of nonlinear functions is typically chosen in the mapping layer; commonly, logistic or tanh is applied.

Researchers have demonstrated that ELM offers superior, or similar, performance as LS-SVMs and SVMs but a much faster training time [55]. This and the limited number of hyper-parameters that need to be selected encouraged Huang to argue that ELM can promote real-time learning where the learning can be conducted without human intervention. The main limitation of ELM is its inability to handle dynamic systems, which prevents its implementation in many real-world applications.

3.3.1.9 *Extreme Kernel Machines*

In the extreme kernel machines (EKM) version of ELM, the input vector $\vec{x}$ is not mapped to a higher dimensional space by a random matrix, but by a kernel. Similar to SVMs, the kernel applies here, which means users do not have to know the actual mapping function. It is important to state the differences in applying the kernel among EKM, SVMs and LS-SVMs. In EKM, the mapping does not depend on the target label as the kernel is applied on the input vector only, which may explain the superior generalisation performance compared to SVMs and LS- SVMs. Another important difference is that SVMs and LS-SVMs are binary classifiers where EKM is not, which allows it to deal with multi-label tasks efficiently. The main equation of EKM used to estimate output function from a training dataset is as follows [57]:

$$f(\vec{x}) = h(\vec{x})\mathbf{H}^T(\frac{\mathbf{I}}{C} + \mathbf{H}\mathbf{H}^T)^{-1}\vec{y} \tag{3.10}$$

$$= \begin{bmatrix} K(x, x_1) \\ \vdots \\ K(x, x_N) \end{bmatrix}^T \left(\frac{\mathbf{I}}{C} + \mathbf{\Omega}_{ELM}\right)^{-1} \vec{y}$$

where $\mathbf{H}$ is a square matrix (N by N), $h(\vec{x})$ maps the input vector $\vec{x}$ to a higher space using a kernel and C is the regularisation parameter. As can be seen from the previous brief mathematical description, users do not need to specify the number of nodes used in the mapping layer; as in EKM, the length of the mapping function is the number of the training sample. In other words, in EKM, the output of the mapping layer cannot exceed the number in the training sample; however, this is not guaranteed in ELM.

3.3.2 *Time Series Classification*

Classification of time series is critical to the design of real world applications, as many of the real world phenomena such as speech and vision are in the form of time series. However, time series classification is much more challenging, as it requires dealing with dynamic systems where, unlike static systems, the output is determined not only by the current input but the previous input as well as the current state will influence the response of the system as well.

The literature is full of attempts that apply learners that know about their lack of knowledge in handling dynamic systems such as support vector machines, Naive Bayes and K-Nearest Neighbour[93]. Although these algorithms can do well on many of the benchmarks because most of these public datasets have a low level of noise, their performance drops dramatically when used for data with added noise[80]. This information is very critical in designing real-world applications as many of these tasks require handling very noisy data[32] [45].

This is can be clearly seen in speech recognition systems as changing the noise level significantly influences the performance of the state-of-the-art techniques [94][37]. Another related application that suffers from the same issue is image classification. In this application, a very small shift in the object location in the image will result in very different input vectors. Many

approaches have been developed to tackle this issue, such as extracting a more robust feature from the input vector (the image) or adopting a brute force method by simply adding random noise to the training dataset to increase the robustness of the system. The computational cost of such a method is very high. Another approach to handle the time series, or any dynamic system in general, is to adopt algorithms that are naturally able to model dynamic systems, such as the recurrent neural networks (RNNs)[38]. The main challenge in designing RNNs is calculating the weight of the system in the training phase due to the lack of an efficient learning algorithm that can backpropagate the error signal through a large number of time steps. This, however, started to change when reservoir computing (RC) proved itself to be a reliable and efficient technique to train the RNN and began to emerge as a new research field. The main focus of this study is to investigate the RC techniques for time series classification tasks.

3.3.3 *Dynamic Machine Learning Algorithms*

3.3.3.1 *Hidden Markov Model*

The hidden Markov models (HMMs) is a probabilistic graphical model (PGM) that can be considered to be an extension to the Markov chains representation. Essentially, HMM is a generative sequence classifier that consists of two sets of variables: observed and hidden. The aim of HMM is to infer the state of the hidden variables based on the data. It has been successfully applied in a variety of real-world applications such as robotic localisation, genome analysis and natural language processing. It makes a very strong assumption, known as the Markov assumption, that enables it to handle large number of variables. In practice, HMM tends to provide a good performance even if this assumption

is violated. This and the fact that HMM can be efficiently trained in supervised and unsupervised modes are its main attractions.

In this section, we formally introduce HMM and discuss the reasons behind its success in the speech recognition domain and its main limitations. Formally, Markov chains can described as follows: Given a set of states $Q = q_1 q_2 \dots q_N$ where N is the number of states, and transition probability matrix $A = [a_{01} a_{02} \dots a_{nn}]$ where each entry of A represents the probability of moving between specific states, and finally, the special start and end state q_0 and q_F that are not related to the observed variables. The Markov assumption is the following:

$$P(q_i|q_1 \dots q_{i-1}) = P(q_i|q_{i-d} \dots q_{i-1}) \tag{3.11}$$

where d is the number of the considered previous states ,the MC order, so for first-order MC model the equation becomes :

$$P(q_i|q_1 \dots q_{i-1}) = P(q_i|q_{i-1}) \tag{3.12}$$

As can be seen from the above, in MC all of the variables are observed, and the complexity of model is controlled by the value of the MC's order. The N-gram language modelling is a typical example of the MC model in the natural language processing domain. Giving a set of words $W = w_1 w_2 \dots w_N$ where N is the number of words in the lexicon, the N-gram aims to estimate the probability of the next word given the N previous words. The transition matrix A is computed by the maximum likelihood estimation algorithm. The size of A grows exponentially with the order of the MC which means this model can only provide a short-term memory.

In HMM the MC is not directly observed (hence, the name hidden), and the model estimates the current state based on other observed data.

Formally, an HMM consists of an MC (defined by Q hidden states, q_0, q_F and A, the transmission matrix); a set of observed variables $O = o_1 o_2 \dots o_T$ where

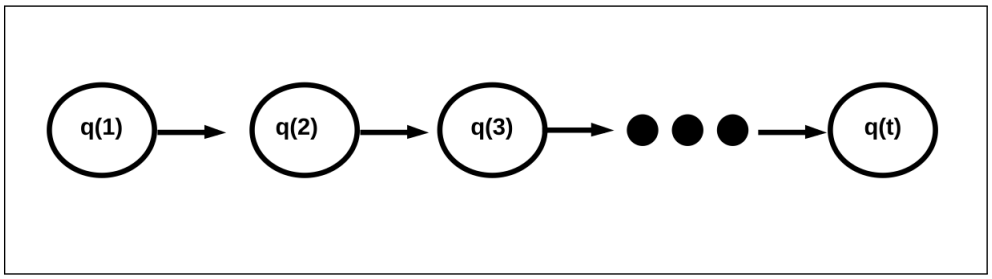

Figure 3.3: A first order Markov chain.

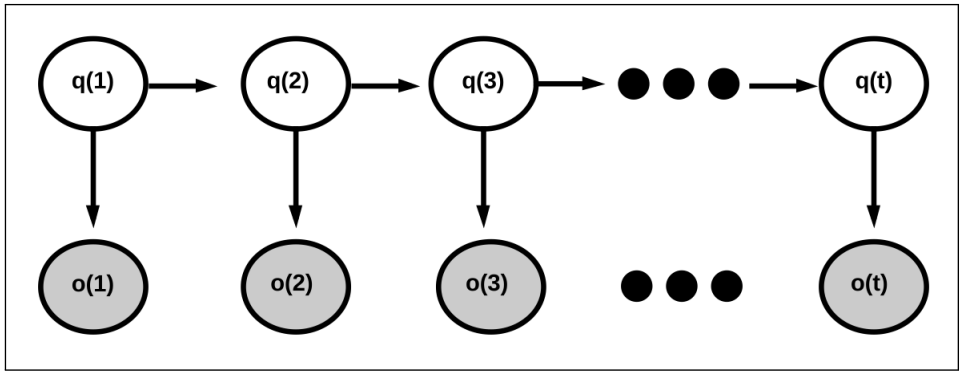

Figure 3.4: A first order hidden Markov model where the observed variables are shaded .

T is the number of the observed variables; an emission probability matrix E (which contains the probability of an observed variable o being generated from a specific state). Giving the HMM model, there are three fundamental problems: likelihood, decoding, and learning.

The likelihood problem is concerned with estimating the probability of a sequence O giving a HMM model $\lambda = (A, E)$. This is a very computationally expensive calculation that is computed by adopting the dynamic programming paradigm, namely, the forward algorithm. By implementing the forward algorithm, the computational complexity of the task is reduced from an exponential term N^T to $O(N^2T)$ where N is the number of the hidden states and T is the length of the observed sequence. This enables the HMM to handle a long sequence of observation in an efficient manner. The task is different in the decoding problem as the aim is to find the best sequence of the hidden state, giving a sequence of observations O and a HMM model $\lambda = (A, E)$.

Dynamic programming is also used here. In particular, the Viterbi algorithm is used to solve this problem.

Finally, learning is the third problem in which the task is to estimate the parameters of the HMM model. In particular, given a sequence of observation O, the aim is to find $\lambda = (A, E)$ that maximises the probability of the data. This training stage can be achieved in unsupervised and supervised modes. In the supervised setup, the target labels are given, making it easy to learn the model's parameters, which is typically achieved by adopting the maximum likelihood algorithm. The learning task becomes more challenging in the unsupervised paradigm in which target labels are missing. An expectation maximisation algorithm is used in this case, namely, the forward-backward algorithm, also known as the Baum-Welch algorithm. It is an iterative method that consists of two-step expectation and maximisation. It starts with a random initialisation of the model's parameters, which are updated in each iteration to obtain a better fit to the data. This algorithm provides a significant advantage for HMM over other discriminative sequences of learners as it enables the use of unlabeled data, which is typically easier to obtain even for large-sized data. However, a main limitation of the Baum-Welch algorithm is that it suffers from local optima solutions.

3.3.3.2 *Time Delay Artificial Neural Networks*

In the previous section, the nature of dynamic systems was discussed, and emphasis was placed on its temporal characteristics. There are two approaches to enable artificial neural networks to handle dynamic systems, and both aim to introduce memory into the model. The first approach is adding feedback to the network, namely, the recurrent neural network, which is described in the next section. The other approach aims to provide the required memory by adding a delay to the network. This section is devoted to this model. It was

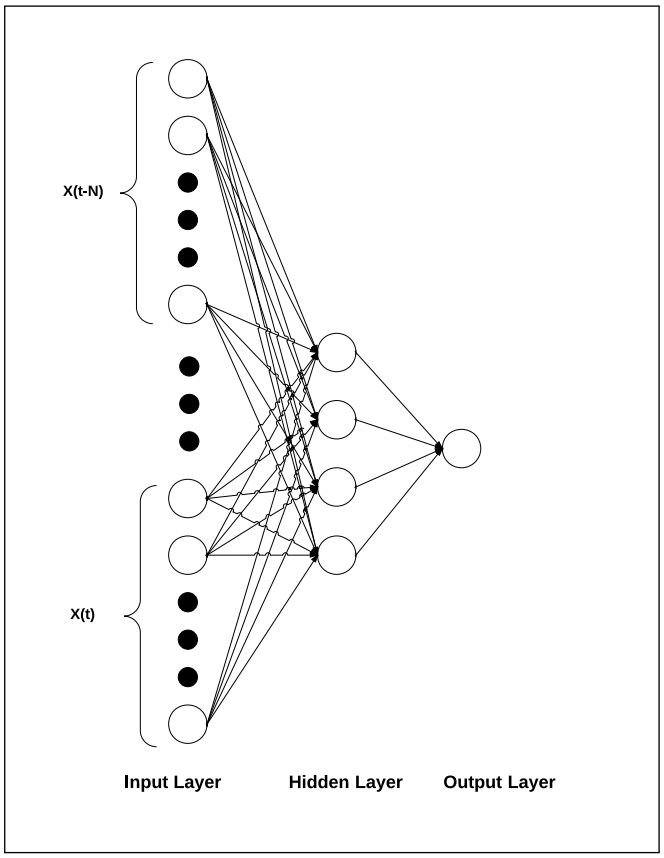

Figure 3.5: A single hidden layer time delay artificial neural networks with an N time delay.

introduced in the 1980s and showed superior performance for many real-word tasks[96].

A major limitation of the Multi-Layer Perceptron model, as discussed earlier, is that it cannot handle dynamic systems. Time delay artificial neural networks overcome this limitation by introducing a delay into the MLP structure. This delay allows the network to capture the temporal information and to provide a memory, in which the number of delays controls the length of the memory. Apart from the added delay, time delay artificial neural networks and MLP are identical, including the use of the same training algorithm (namely, back-propagation) in the learning phase. This also means that both models share the same limitations that are related to the training phase, including local minima and hyperparameters.

In addition, in a regime with a long memory, in which the network needs to maintain information from many past steps, it requires the dimension of the input layer to increase dramatically. This is particularly true in high-dimensional data wherein each time step includes a large number of features. A good example can be seen in a scene detection task in which the each time step is an image of 800 by 600 pixels, which requires 480,000 features for each time step. Suppose that a 10-step time delay is needed; then, the total number of the input layer increases to 4,800,000 features. Training such a network is very computationally expensive and tends to have a high variance due to the increase of the model complexity (number of weights).

3.3.3.3 *Recurrent Artificial Neural Networks*

In the previous section, we discussed that, in modelling dynamic artificial neural networks, a type of memory is required to capture the temporal structure, and one of the two main methods, namely, time delay artificial neural networks, was described. In this section, we shift the focus to recurrent artificial neural networks, another approach to embedding a memory into the network structure. The recurrent architecture is not a recent introduction to the field. In fact, it was heavily discussed in the literature[68]and[83]. It is more biologically plausible than the feedforward network, and the theoretical analysis shows a great potential for adopting the recurrent network.

Despite all of these attractions, the use of the recurrent neural network in real-world applications remains very limited. This is mainly due to the absence of an efficient learning algorithm. The research community has tackled this problem by extending the training algorithm of feedforward network, the backpropagation algorithm, to the recurrent structure. This results in the development of the backpropagation through the time algorithm, in which each time step is considered a different layer and the error is backpropagated through these time steps. This, however, has all the limitations of the back-

propagation algorithm (discussed in section3.3.1.5) in addition to a serious problem in the training model that requires long-term memory as the gradient tends to vanish. This is known as the vanishing gradient problem.

To address this problem several recurrent neural structures have been developed. The first proposed recurrent neural structures found in the literature are the Elman networks and the Jordan networks[28]. Both networks share a recurrent layer that simply copies the current state value to the next time step, known as the context layer. The backpropagation algorithm is used in the training phase as in the feedforward network setup. The main difference is in the position of the context layer. In Elman networks, the hidden layer is selected, whereas the output layer is chosen in Jordan networks. The vanishing gradient is not a serious problem here as the error is typically propagated over three layers (a single hidden network). Due to this simple architecture, these nets are known as the simple recurrent network (SRN). Because it can be trained efficiently, this simple structure is effective in a regime that does not contain long-term dependencies.

Another network was introduced in[51] in the late-1990s, namely, long-short-term memory (LSTM). This network overcomes the limitation of the SRN in learning long-term dependencies. The main novel concept in this net is the constant gradient flow that allows the network to avoid the vanishing gradient problem. This is achieved by introducing special units (neurons) with gates to control the value of the gradient, known as the memory cell. In practice, it is widely considered to be the most successful approach to extend the backpropagation algorithm to the recurrent net. It is the state-of-the-art learner for many tasks, such as handwriting recognition [38]. Despite this significant success, this net suffers from two main weaknesses: the high computational cost in the training phase (it is not uncommon to train the system for weeks before obtaining a good performance [21]) and a tendency towards overfitting. It is common to attempt to avoid the latter problem, high variance, by using a

large dataset, typically by adding a type of distortion, but this increases the training time.

Recently, novel RNN topologies have been introduced based on the reservoir computing concept. These methods can capture long-term dependencies while being very fast to train. This novel concept and its main technologies are discussed in depth in the remainder of this chapter.

3.3.3.4 *Reservoir Computing*

Reservoir computing is an emerging field that offers a novel approach to training recurrent neural networks. It was developed in 2002, and since then, its popularity has grown rapidly due to the simplicity of its implementation and the robust performance[95]. RC contains several techniques that have been derived from different backgrounds. However, all of them share the main idea that consists of the random initialisation of the weight of the recurrent nodes and only learns weights in the output layer by using simple readout functions. The two major approaches that lie under the umbrella of RC are the echo state network (ESN) and the liquid state machine (LSM).

3.3.3.5 *Echo State Network*

ESN was introduced by Jaeger in 2001[59] and has been applied to different real world applications where it proved to achieve a superior performance, or similar, compared to the state-of-the-art algorithms. This success has led to a wide acceptance of this technique in the field and encouraged researchers to conduct studies that aim to explore the fundamental properties and behaviour of ESN that lies behind its high performance. Another, rather more empirical effort has also been made to investigate the applicability of ESN on new and more challenging real world problems and to conduct extensive comparisons among the state-of-the-art techniques [70]. The ESN model is characterised in the following way. First, $\mathbf{W^{in}}$, which is an m by n matrix (where m is the size

of the input vector and n is the size of the reservoir), is initialised randomly. Second, $\mathbf{W^{res}}$, which is an n by n matrix, is initialised randomly as well and scaled to obtain the desired dynamics. Another important component of this model is the fading memory (forgetting) parameter α , which plays a major role in controlling the memory capacity of the reservoir. The model update equations are as follows[95]:

$$\bar{x}(t) = f(\mathbf{W^{in}}[1; u(t)] + \mathbf{W^{res}}x(t-1)) \tag{3.13}$$

$$x(t) = (1-\alpha)x(t-1) + \alpha\bar{x}(t) \tag{3.14}$$

where, $u(t)$ is the input signal at time t and f is a nonlinear transfer function, commonly logistic or tanh. The response of the reservoir dynamic and the class labels of training are used to train a simple linear read-out function that results in learning the weight of the output layer $\mathbf{W^{out}}$. This is typically accomplished by applying the pseudo-inverse equations, as follows:

$$\mathbf{W^{out}} = (\mathbf{X^T_{response}X_{response}})^{-1}\mathbf{X^T_{response}Y} \tag{3.15}$$

where $\mathbf{X_{response}}$ is an n by p, which is the size of the training set, matrix that contains the response of the reservoir and $\mathbf{Y}$ is an p by c, which is the number of different classes, matrix that encodes the target labels.

Hyper-parameters Optimisation

The ESN hyperparameters include the size of the reservoir, the leakage rate, the input scaling factor, the reservoir scaling factor, the applied nonlinearities and the regularisation coefficient. There is not an agreed-upon approach to optimising these parameters, but a common practice is to use a validation set to

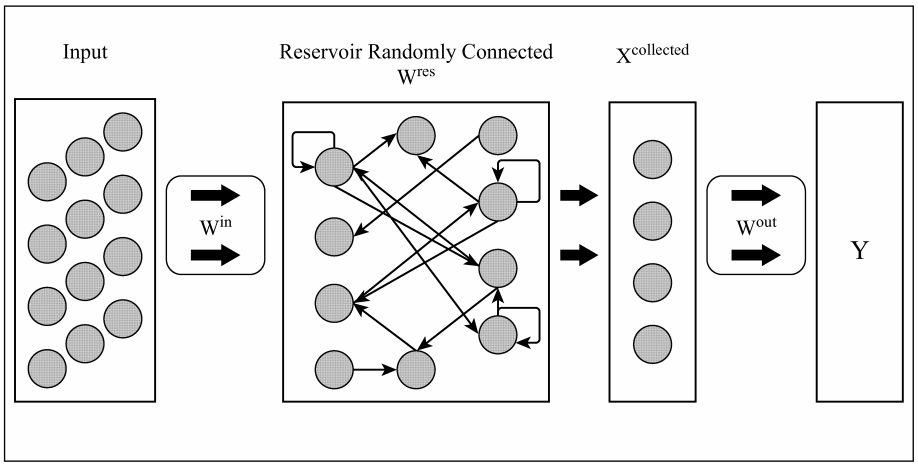

Figure 3.6: The structure of the ESN and readout system. On the left, the input signal is fed into the reservoir network through the fixed weights $\mathbf{W^{in}}$. The reservoir network recodes these, and the output from the network is read out using the readout network on the right $\mathbf{W^{out}}$,which consists of the learnt weights.

perform a grid search. The importance of each parameter differs significantly as the size of the reservoir and the leakage rate have the highest impact on the performance. The accuracy of the system tends to improve as the reservoir size increases, assuming that overfitting is avoided by effective regularisation. Thus, it is common that the computational cost and the hardware limitation dictate the selection of the reservoir size. Controlling the model memory is achieved by changing the leakage rate, which needs to be optimised to capture the data temporal structure. The scaling coefficients for the input and the reservoir are strongly related; thus, they are typically optimised together. In the nonlinearities, sigmoid functions, particularly tanh or the logistic function, are used but in principle any nonlinear function can be used as, unlike the error propagate method, the function is not required to be differentiable. The regularisation coefficient is optimised using the validation set. As can be seen from the previous description, optimising the network is a very complicated process that requires experience in manual fine-tuning such models to detect classic problems such as high bias or high variance, under- or overfitting. An

automated optimisation paradigm is still missing in the literature. Limited or known human intervention is needed with the exception of pioneer work in[31] and[18] that aims to evolve the network by adopting evolutionary methods.

3.3.3.6 *Liquid State Machine*

LSM has been developed from a neuroscience background and was introduced by Maass in 2002[72]. The aim of LSM was to simulate the behaviour of neural systems, which might explain its limited adoption in real-world applications compared to ESN. Maass introduced LSM as a novel biological computational model that, under ideal conditions, guarantees a universal computational power. He placed significant emphasis on the previous statement and showed that LSM could emulate the Turing machine. The major difference between ESN and LSM is the type of nodes, as in LSMs spiking nodes are adopted, unlike in ESN. Commonly, leaky integrate-and-fire models are applied in LSM, but several attempts have been made in the literature to use more realistic models[97]. In summary, LSM offers a new perspective for understanding and modelling the brain as a liquid that responds differently based on the excitement type and to use the liquid response to train a simple readout function.

Despite the differences in the objectives of ESN and LSM, there is a crucial need to perform an extensive experiment to determine the strength of each technique; in particular, what can be achieved by applying LSM over ESN in the context of real-world applications and whether each approach may be more suited to a specific kind of application, e.g., speech processing or computer vision. These questions have not been addressed in the literature although the answers will provide a greater insight into the underlying nature of each technique and RC in general.

3.3.3.7 *Attractions of Reservoir Computing*

There are several attractions of RC over the traditional approaches in modelling dynamic systems. These attractions stem from two main sources: The first emerges from the fact that RC is an RNN, which means that it offers all of the attractions of RNNs. This includes the ability to model a time series without discrete states, which is widely considered the major attraction of RNNs as it gives the model an advantage over such state-of-the-art techniques as the Hidden Markov Model (HMM). HMMs rely heavily on the transitions between discrete states of the systems, and the absence of such information to facilitate even a poor estimation of the emission matrix severely affects performance. A good example can be seen in the task of recognising musical instrument sounds where the model does not have discrete states, which simply prevents the use of HMMs. Speech recognition is another example of the limitations of HMMs versus RNNs because the language model applied to the system plays a major role in influencing performance[60]. In other words, RC (and RNN in general) covers a broader range of tasks where the models do not have discrete states and avoids all of the issues related to calculating the probabilities of the transactions among the model states and the emission matrix. It is important to state that such knowledge can be embedded with RC to implement a maximum entropy model that combines the predictions of the model with the emission matrix and uses dynamic programming approaches such as the Viterbi algorithm to estimate the most likely sequences.

The second source of attraction arises directly from the underlying nature of RC versus other RNN training algorithms. The literature describes several approaches to training RNN, including back-propagation through time (BPTT) and long-short term memory (LSTM)[51]. However, these methods are computationally expensive and suffer from a variety of issues. BPTT cannot handle the vanishing gradient problem, which limits the use of RNNs and deep neural

networks. In LSTM, the high computational cost, the need to have a very large sample and the tendency of such big RNNs to over-fit remain serious barriers to its wider adoption in the research community. On the other hand, RC overcomes the limitations of traditional methods as it can be modelled using a very large number of steps without suffering from the vanishing gradient; as RC does not propagate the error, the weights of the recurrent nodes are set randomly. In addition, RC is extremely fast compared to LSTM as only the weights of the final layer are included. Another advantage of the RC approach is that multi-task systems as reservoirs can be used as mapping functions and different teaching signals (labels) can be used on the final layer to construct several linear read-out functions, one for each task. Multi-task systems are not enabled with traditional training algorithms, which gives RC another major advantage over them.

4

RESERVOIR COMPUTING FOR ARABIC SPEECH RECOGNITION

4.1 INTRODUCTION

Reservoir Computing has recently been applied to speech recognition with impressive success in the English language domain [94]. This success encouraged us to investigate the benefits associated with adopting reservoir computing methods when developing an Arabic speech recognition system. Resources in Arabic are limited, which creates in itself a significant challenge for this study. Many algorithms can perform well when trained on a large corpus but their performance drops dramatically when they are applied to a far smaller corpus. In other words, the aim of this chapter is to explore whether the same success reported in the English domain can be achieved with a more limited resource language, namely Arabic. In order to achieve this aim, we have investigated different pipeline architectures based on the echo state network. Developing several architectures aimed to address the issue that classification techniques tend to be sensitive to the preprocessing phase. Thus, several preprocessing methods were considered in this study to ensure that valid results were produced. To the best of our knowledge, this is the first attempt in the literature to adopt reservoir computing for the Arabic language, which has then been followed by other efforts in the literature [50]. This chapter is organised as follows. The related work is firstly discussed in the second section to establish the context for our efforts among the large research community and provide a basis for comparing the scope of our study with others found in the literature.

The corpora used in this study are described in the third section, including our own developed corpus which is the largest of isolated Arabic words. This covers the motivation behind its development, the development process and the corpus description. The systems' architectures are discussed in the fourth section together with the related technical details such as the software used and the hyper-parameters optimisation process. In the fifth section, the results are stated and analysed, and comparisons are carried out among the different systems found in the literature. The limitations and potential architectures are discussed in the sixth section and finally a conclusion is drawn in the seventh section.

4.2 RELATED WORK

The Arabic language is the official language of 26 countries and among the six official languages of the United Nations. It is spoken by over 300 million people, which makes it one of the 10 most widely-spoken languages in the world [20]. Despite that, the literature on automated Arabic speech recognition is limited compared to other languages. This includes a shortage of conducted studies, and a lack of depth in these studies and public resources. The lack of benchmarks has led researchers in the field to rely on building an in-house corpora, which is an expensive and time consuming process. Moreover, these corpora tend to be small and not shared within the community, preventing researchers from making valid comparisons between the published systems. The only publicly accessible corpus (SAD) is published in a processed format (as MFCCs) which prevents the development of novel feature extraction methods and comparisons between them. This duplicated effort can be avoided by providing public corpora such as the one we introduce here (available at http://www.cs.stir.ac.uk/~lss/arabic)

Systems	Corpus	Feature Extraction Method	Classification Method
Hu et al. [53]	(SAD) 8800 utterances	MFCCs	Wavelet Neural Networks
Hammami and Sellam [42]	(SAD) 8800 utterances	MFCCs	Tree distributions approximation model & HMMs
Elmougy and Tolba [29]	600 utterances	MFCCs	Ensemble/ Multi Layer Perceptron (MLP)
Alotaibi [6]	1700 (digits) & 4000 (vowels) utterances	MFCCs	ANN & HMM
Astuti et al. [9]	2000 utterances(digits)	MFFCs	Suport Vector Machines (SVMs)
Ali et al. [5]	300 utterances	MFCCs	Multi Layer Perceptron (MLP)
Hammami et al. [44]	(SAD) 8800 utterances	MFCCs	Copula Probabilistic Classifier
Ganoun and Almerhag [35]	130 utterances	MFCCs & Walsh spectrum & Yule-Walker	Dynamic Time Warping (DTW)
Hachkar et al. [39]	500 utterances	MFCCs	Dynamic Time Warping (DTW) & DHMMs
Hachkar et al. [40]	2700 utterances	MFCCs & PLP	HMMs
Nadia Hmad and Tony Allen [50]	3802 utterances (phonemes)	MFCCs & LPC	ESN

Table 4.1: A summary of the proposed systems found in the literature.

To identify the current state of the field, including the existing challenges, a brief literature review has been conducted. A summary of the systems found is shown in Table 4.1, which contains some of the proposed systems published over the past 5 years. The main observation is the common use of an in-house small corpus, with the exception of (SAD). In addition, the majority of the studies focus on the classification stage, and limited attention has been directed towards the investigation of the feature extraction methods, although this step has a significant impact on system performance[99]. MFCCs are adopted as the feature extraction method in almost every proposed system. A variety of classification techniques has been applied, but HMM and MLP are the most widely-adopted approaches. The results of many of these systems cannot be compared as they have been reported only for private corpora.

The most closely-related work found in the literature is[50], which was published after our work in [2]. It is important to state that neither researcher was aware of the other's efforts before these two papers were published. We have contacted the authors to discuss a possible collaboration which would result in exchanging corpora and other resources. This would allow us to make a direct comparison among these two studies (see the next chapter for the details). Although both studies aim to develop a reservoir computing-based Arabic speech recognition, there are many significant differences between them. The main difference is that, in[50], an Arabic phoneme recognition system was developed instead of a word-based system. In order to achieve this, a phoneme corpus has been developed using CSLU2002 continuous speech. The developed corpus has been developed manually and only a subset of the CSLU2002 corpus has been used. This subset contains 3802 utterances of Arabic phonemes spoken by 34 speakers (17 females and 17 males). Two feature extraction methods were considered in the cited study, namely MFCC and LPC. Despite the importance of this paper, being among the first attempts to apply ESN to the Arabic domain, there are several limitations to this work.

The main shortage of this work is the absence of any comparison with other work in the literature, that uses different approaches, which is critical for evaluating the developed system performance; however, this may be explained by the absence of an Arabic corpus. As stated above, the authors developed the phoneme corpus, which meant that comparisons with previous work were impossible. The issue could have been addressed by comparing the developed model with a baseline model but even this approach has not been followed. In addition, the developed corpus is too small to draw any robust conclusion. Despite that, the discussed work is considered to be a very important effort towards advancing speech recognition systems for the Arabic domain.

In summary, the main challenges found in the literature fall into three areas. Firstly, there is a need to introduce public corpora and encourage researchers to share their resources in order to ensure that valid comparisons are made between the published systems. Secondly, more focus needs to be directed towards the feature extraction methods in developing novel systems. Finally, noise robust systems that can be applied in real-world applications need to be proposed.

4.3 CORPORA

In this section, we describe the corpora used to test and evaluate the developed systems. Three corpora have been used in this study, namely the Arabic Phonemes Corpus, the Spoken Arabic Digit (SAD), and our own developed corpus - the Arabic Speech Corpus for Isolated Words. Only the latter two corpora are publicly accessible, while the first (the Arabic phonemes corpus) has been obtained from the authors based on ongoing collaboration on condition that it will not be distributed. The strengths and limitations of each study will be discussed in detail, combined with the related studies that have been conducted

on these corpora. We will structure this section based on the release date of the each corpus, starting with the corpus that was introduced first: SAD.

4.3.1 *The Spoken Arabic Digit Corpus (SAD)*

The Spoken Arabic Digit Corpus is by far the most widely-cited corpus with regard to Arabic speech recognition systems. It was developed in 2008 at the Laboratory of Automatic and Signals, University of Badji-Mokhtar, Algeria. This corpus contains the Arabic digits from 0 to 9 uttered 10 times each by 88 native Arabic speakers (44 females and 44 males). This means that it contains 8800 samples which have been divided into two separate subsets: one for training which contains 6600 samples and the other for testing, contains the remaining 2200 samples. The speakers in the training set are not the same speakers in the test set. This corpus is only available in a preprocessed format (MFCCs) which has been computed using the following parameters: sampling rate 11025 Hz with 16 bits resolution, hamming window and a filter of $1 - 0.97z^{-1}$.

Since its release, several studies have used this corpus to evaluate and compare systems[41][16]. This popularity arose due to several factors, including the ease of obtaining this corpus (available in a pre-processed format and so ready to use) and the relatively large number of speakers which allows researchers to train complex systems. All of these attractive properties have contributed to the wide adoption of this corpus in the research community. This corpus, however, suffers a serious limitation, the absence of the recordings in their raw format. This has prevented scholars from investigating other feature extraction methods and constrains all of the conducted studies that have used this corpus to using the MFCCs approach. Adding noise to mimic real-world scenarios where a high level of noise tends to be present or testing the system under different environments are also impossible. In addition, although the dataset

contains a large number of speakers, the vocabulary size of 10 words, is too small. In spite of that, this corpus has continued to be one of most importance resources for the Arabic language.

As stated above, many studies found in the literature use this corpus. All of these studies focus only on the classification stage and pass the MFCCs vectors directly to the classifiers or compute the dynamic features by calculating the first and second derivatives of these vectors [42][16].

4.3.2 *The Arabic Speech Corpus for Isolated Words*

An Arabic Speech Corpus for Isolated Words has been developed by the author at the Department of Management Information Systems, King Faisal University. It contains about 10,000 utterances of 20 words spoken by 50 native male Arabic speakers; Table 4.2 shows the selected words. The corpus has been made freely accessible for non-commercial use in raw format (.wav files) and other formats to allow researchers to apply different feature extraction methods. The corpus has been recorded with a 44,100 Hz sampling rate and 16-bit resolution, as well as a two channel-stereo mode. Each of the files has been labeled using the following coding system:
S (Number of Speakers).(Number of Repetitions).(Number of Words)

The following is an example: S01.01.01. It represents the first of the 50 speakers, the first recording of the 10 and the first word from the list of 20 words. This coding system allows researchers to use this dataset not only for speech recognition systems but also for different classifications tasks, e.g., speaker identification systems. In order to ensure a valid comparison between the developed systems, the dataset has been divided into two sub-datasets one for the purpose of training and parameter estimation and the other for testing. The training dataset contains 80% of the total dataset (7,993 samples), and the

Arabic	Translation	English Approximation	IPA	Number of Utterance
صفر	Zero	Safer	/ sˀfr /	93 utterances
وَاحد	One	Wahed	/ waːhid /	100 utterances
اثنَان	Two	Ethnan	/ ʔθnaːn /	100 utterances
ثلَاثة	Three	Thlatha	/ θalaːθh /	100 utterances
أربعة	Four	Arbah	/ ʔrbaʕh /	100 utterances
خمسة	Five	Khamsah	/ xmsat /	100 utterances
ستة	Six	Setah	/ sitat /	100 utterances
سبعة	Seven	Sabah	/ sabʕah /	100 utterances
ثمَانية	Eight	Thamanah	/ θmaːnjh /	100 utterances
تسعة	Nine	Tesah	/ tisʕah /	100 utterances
التنشيط	Activation	Al-tansheet	/ aːtanʃytˤ /	100 utterances
التحويل	Transfer	Al-tahweel	/ aːtaħwyl /	99 utterances
الرصيد	Balance	Al-raseed	/ aːrasˀyd /	100 utterances
التسديد	Payment	Al-tasdeed	/ aːtasdyd /	100 utterances
نعم	Yes	Naam	/ nʕm /	100 utterances
لَا	No	Laa	/ laː /	100 utterances
التمويل	Funding	Al-tamueel	/ aːtamwyl /	100 utterances
البيَانَات	Data	Al-baynat	/ aːlbayanaːt /	100 utterances
الحسَاب	Account	Al-hesab	/ aːlħisaːb /	100 utterances
انهَاء	End	Enha	/ ʔinhaːʔ /	100 utterances

Table 4.2: All the words that have been included in the corpus with the number of the utterances for each word and its English approximation and translation.

test set contains the remaining 20% (1,999 samples taken from the second and ninth repetition of each speaker).

4.3.2.1 *Corpus Generation*

Developing the Arabic speech corpus for isolated words is a time-consuming task that consists of several challenges. In this section we describe the generation process of this corpus. A significant amount of attention has been devoted to the planning phase to ensure that all the aims are clear and realistic. This includes the list of words and the size of the corpus. The size of the corpus has been selected to ensure that this corpus is the largest in its category. The list of words has been selected to be representative of real-world scenarios. The financial sector has been chosen as the focus of this corpus due to the relatively limited words needed in such applications and the familiarity of its words to the prospective participants. This focus helps to reduce mistakes during the recording time and ensures a higher level of positive response in the searching process for participants. The equipment and the software required have also been determined during the planning phase. The equipment includes a MacBook Pro (Processor 2.7 GHz Intel Core i7, RAM 8 GB) and a Snowball microphone (CD quality). The free digital audio editor Audacity has been chosen for use in the recording and the preparation process. The planning process also covers the selection of the location in which the corpus will be collected. This location needs to be suitable for conducting the actual recording in terms of noise level and equipment and needs to be a close distance from prospective participants. King Faisal University agreed to host the data collection process and to provide us with revenue that met the required conditions. The data collection process consisted of searching for prospective participants and describing the recording process for them. Each participant was given the list of the selected words and was asked to pause after each word for four seconds to enable the separation of the words. Once the acoustic

signal had been recorded, each word was extracted and named in the system described in the previous section. All of these files have been played twice to ensure the quality of the signal and that the correct label was used.

4.3.3 *The Arabic Phonemes Corpus*

The Arabic Phonemes Corpus is a recently-developed, private corpus that aims to provide a phonemes-labelled corpus for the Arabic language. It has been developed at Nottingham Trent University by Nadia Hmad and Tony Allen[50]. A subset of the CSLU2002 corpus, which is a large commercial continuous speech corpus that covers 22 languages including Arabic and has been recorded with 16 bit resolution and sampled at 8 kHz, has been selected, manually-segmented and labelled, which is an expensive process. This corpus contains all 33 Arabic phonemes uttered by 34 native Arabic speakers (17 females and 17 males, selected randomly from the 98 speakers of the CSLU2002 corpus). It contains 3,802 samples that are divided into two subsets: one for training (1,894 samples) and the remaining 1908 samples for testing. These samples are not equally distributed over the phonemes and the phoneme with the lowest number of samples has 76 instances whereas the average sample for each phoneme is 114 and the maximum is 120. The primary strength of this corpus is that it is phoneme-labelled manually which, unlike other corpora, allows researchers to develop a phoneme-based speech recognition system. On the other hand, its weaknesses include that it has too few speakers and samples per phoneme. Also, it has not been released publicly which prevents other researches from making valid comparisons and limits the reproducibility of the reported results. This corpus has not been used in the literature, except for the above cited article that introduces it and uses it to evaluate a speech recognition system, mainly due to the fact that it has recently developed and has not been publicly available.

In this section, we report the conducted experiments on the considered corpora. The scope and limitations of these experiments and their findings are also discussed. It is important to start this section by restating the aim of this chapter which shapes the design of these reported experiments and provides a clearer big picture for this work before diving into the details of each individual experiment. The aim is to investigate the potential benefits of adopting ESN in developing an Arabic speech recognition system. In order to achieve this aim, several speech recognition systems based on ESN have been developed and compared to other reported results in the literature and baseline models, where possible, in relation to the three considered corpora.

4.4.1 *Experiments on the Spoken Arabic Digit Corpus*

The Spoken Arabic Digit Corpus was the first to be used to evaluate the developed systems. There are many studies found in the literature that adopt this corpus which allows us to compare the achieved results with those of the reported systems. In these experiments, as the corpus is only available in MFCCs format, we could not evaluate other feature extraction methods, which limits the scope of these experiments to investigating the classification phase alone rather than the complete the speech recognition system. This restriction is not present in the other considered corpora. Some of the results presented here have been published in[2].

4.4.1.1 *Hyperparameters Optimisation*

In order to optimise the hyper-parameters of the developed system, the corpus has been divided into three subsets. The training set, that contains 6,600

instances, has been segmented into two sets one for training, containing 5,000 samples, and the other for validation, which contains the remaining 1,600 samples. Once the hyper-parameters have been optimised, the system is evaluated on the unseen test set which we report in the results section. The Hyper-Parameters of the ESN that need to be tuned to fit the task at hand include the reservoir size, the leakage rate (control of the fading memory), the input scalers and the internal scaler. Due to our limited computational power, the largest reservoir constructed has 1,000 nodes whereas a grid search has been conducted to optimise the rest of the parameters. Matlab code has been written to implement the echo state network following the description in [69].

4.4.1.2 *Results*

The results, summarised in 4.3, show the superior performance of ESN-based systems compared to the recently-reported systems in the literature that uses the same corpus. The comparisons among these studies are valid as the same sets, created by the developer of the corpus, are used to train and test the system. In addition, no extra information beyond the MFCCs has been sent to the classifier. Achieving this improvement in performance while the average accuracy of the compared systems was very high is very encouraging, which prompted us to investigate the developed system further.

Systems	Accuracy Rate
TM (Nacereddine Hammami et al, 2011)[41]	94.04%
LoGID (Paulo R. Cavalin et al,2012)[16]	95.99%
Echo State Network (This work)[2]	96.91%

Table 4.3: The results obtained by the ESN system and from the two compared studies

4.4.1.3 *Discussion*

The developed system shows superior performance to the other systems found in the literature. This promotes the wider adoption of ESN in designing Arabic speech recognition systems. Taking into account the limited resources used to implement and train the proposed system, it is clear that ESN can handle such regimes very well and outperform the other established systems. Some limitations remain for the corpus, however, as the vocabulary system is too small to draw a robust conclusion. In addition, it is unclear how the developed system reacts in the presence of noise or where there is a mismatch between the environment in which the training and test sets were recorded. All of these issues need to be addressed in order to develop systems that can be successfully deployed in the real-world.

4.4.2 *Experiments on the Arabic Speech Corpus for Isolated Words*

In these experiments, the scope has been extended as the newly-developed corpus allows more in-depth investigation. Many of the limitations of the previously-discussed experiments have been overcome. This includes comparing different feature extraction techniques, namely PLP-RASTA, PLP and MFCCs, to test different potential system architectures. As this corpus has been created during this study, evaluating the developed systems with other systems reported in the literature was impossible. Thus, a baseline model based on HMMs was built for comparison purposes. Some of the results of these experiments have been published in[3].

4.4.2.1 *Hyperparameters Optimisation*

In order to optimise the system parameters for each model, the training set (that contains 7,993 samples) was divided into two subsets: one for training

and the other as a cross-validation set with 6,000 samples for training and the remaining 1,993 samples for testing. In the RC model, a grid search has been applied using these two subsets to find the optimal values for the scaling factors for $\mathbf{W^{in}}$ and $\mathbf{W^{res}}$ to derive the reservoir to the desired behaviour. Another rather crucial parameter is the leakage rate that significantly affects the performance of the model, which also has been optimised using a grid search. Finally, the number of nodes and reservoir size were selected empirically as the model and evaluated using different values for 100 to 1500 nodes. The best performance was achieved with 1,500 nodes. It is important to note here that, due to the hardware limitations, we were unable to experiment with larger reservoir sizes, although the system performance did not stop improving in terms of the accuracy rate and the standard deviation.This procedure was conducted for each feature extraction method (MFCCs, PLP and RASTA-PLP). This suggests that it is possible to achieve higher performance using a larger number of nodes; however, we emphasise preventing any over-fitting of the training data (due to the use of such a large reservoir), which will lead to poor performance on tests with unseen data.

In HMMs, the same two subsets have been used to find the optimal number of states and number of iterations for each of the feature extraction techniques. A number of states between 2 and 30 has been considered in the research, and the optimal number of states was found to be 25 for all of the models as a larger number of states reduces system performance. In MFCCs and PLP, the optimal number of iterations is six, whereas in RASTA-PLP it is four.

4.4.2.2 *Evaluation & Implementation*

The system has been evaluated on the test set that was not used in the training phase or in optimising the system's hyper-parameters. This set contains 1,999 samples. In the RC models, the evaluation was conducted 10 times to ensure a valid result and as a counter to the inherent stochastic behaviour due to the

		Preprocessing		
		MFCCs	PLP	RASTA-PLP
Classification	HMMs	97.65%	98.45%	98.8%
	ESN	98.97% (0.15)	99.16% (0.11)	99.38% (0.11)

Table 4.4: The results obtained by the HMMs and ESN with all the considered feature extraction methods. In ESN, we report the mean over 10 runs and the standard deviation.

random initials of the network weights. Here, we report the mean of these 10 runs and the standard deviation. The same test set was used to evaluate HMMs models that were developed with the different feature extraction methods.

Matlab code has been written to implement the echo state network following the description in [69]. The HMMs and feature extraction methods that were implemented in [27] and [66] were adopted.

4.4.2.3 *Results*

The results, summarised in Table 4.4, show the performance of the six models that were built to conduct the comparison between the two classification approaches, namely, ESN and HMMs, with respect to the three feature extraction techniques, namely, MFCCs, PLP and RASTA-PLP. ESN outperformed the HMM under all of the considered feature extraction approaches, which encourages us to promote the use of ESN in automated speech recognition systems in the Arabic language domain. There were differences in performance across the considered feature extraction methods, but RASTA-PLP achieved the best performance with HMM and ESN, indicating the robustness of this feature extraction approach. The overall best performance was achieved by combining RASTA-PLP with ESN.

Despite the crucial importance of the performance element in conducting such comparisons, it is important to consider aspects beyond it. In ESN (and RC in general), the development of systems that can solve more than one problem (such as systems that can perform speech recognition task and speaker identification task at the same time) is one of its main strengths. This flexibility is not present in HMMs, and each model needs to be developed to tackle a single task. In addition, RC covers a broader range of tasks in which the models do not have discrete states and avoids all of the issues related to calculating the probabilities of the transactions among the model states. On the other hand, the robust performance of RC and its flexibility comes at a high cost, as a large reservoir size (large memory cost) is required in order to achieve a state-of-the-art performance.

4.4.2.4 *Discussion*

A novel speech recognition model based on RC and RASTA-PLP was proposed and evaluated using a newly-developed corpus, which is for non-commercial use. This corpus contains approximately 10,000 utterances of a list of 20 words uttered by 50 native speakers. Several feature extraction methods were compared on the same corpus, namely, MFCCs, PLP and RASTA-PLP. An HMMs model was used as a baseline, and the proposed system achieved higher performance under all of the feature extraction approaches. Future work will include evaluating the system's robustness in noisy environments, see Chapter five. This is particularly important for real-world applications as the signal tends to be noisy and conventional methods, such as HMMs, are known for their poor performance in such environments. In addition, an investigation of the system's usability in Arabic continuous speech and the possible use of a language model will be conducted. Finally, we will seek international cooperation to develop a new public corpus of Arabic continuous

speech that will serve as a benchmark and will encourage the development of new systems.

4.4.3 *Experiments on the Arabic Phonemes Corpus*

The Arabic Phonemes Corpus is, unlike the previous discussed corpora, phoneme-labelled. This allows us to investigate the potential gain in building a phoneme-based speech recognition system. In other words, the aim of these experiments is to evaluate the adoption of the ESN technique in developing a phoneme-based speech recognition system rather than a word-based system. This involves several challenges; the first is that phonemes are far shorter than words, which must be considered in the design of the system. The number of classes is significantly larger than the other two considered corpora as it has 36 classes whereas the SAD has 10 classes and our corpus has 20 classes. The size of this corpus is very small, and it is the smallest corpus among the considered corpora. This means that it has the larger number of classes and the smallest number of samples which results in a very limited number of instances per class (an average of 120 samples per class). All of these constrains imposed by this corpus create a serious challenge in designing speech recognition systems. This is reflected by the achieved performance across all of the systems employed in this study and that found in the literature, and the best system achieves the lowest performance among the three corpora.

In order to achieve the aim of this study, a phoneme speech recognition system, based on ESN, has been developed. Two different feature extractions have been considered, namely PLP and MFCCs, but the use of RASTA-PLP is impossible as the number of frames per phoneme is too small to compute the RASTA filter. In this study, we have also considered two different activation functions (sigmoid and tanh). This means that four different architectures have

been implemented in this study. To evaluate the performance of these systems, we compare them to the work found in the literature[50].

4.4.3.1 *Hyperparameters Optimisation*

This corpus has been divided by its developer into two sets; one, for training, contains 1,894 samples, and the other, for testing, contains 1,908 samples. To ensure a valid comparison with the published work, we used the same two sets as in [50]. In order to optimise the hyper-parameter parameters of the developed systems, we segmented the training set into two subsets: one for training (75% of the samples) and the other for cross validation (the remaining 25% of the samples). We maintained the same reservoir size used in [50] to ensure a valid comparison between the developed systems. Once the hyper-parameters have been optimised, we test the developed systems on the unseen test sets. We run the test 10 times and report the mean standard deviation to account for the randomised elements in constructing the ESN-based systems. Matlab code was written to implement the developed systems based on the description stated in [69] and the library [27] was adopted for the computations during the feature extraction phase.

4.4.3.2 *Results*

In this section, the results of the conducted experiments on the Arabic Phonemes Corpus are discussed and compared with the systems found in the literature. The results of the four developed systems are summarised in Table 4.5. It is clear that the activation and feature extraction methods influence system performance, and that adopting the PLP approach in the preprocessing stage improves system performance regardless of the activation function. This does not hold true for the activation function as no single activation function is superior, regardless of the applied feature extraction methods. The use of the tanh improves performance when combined with PLP and degrades it

when combined with MFCCs. In addition, systems that adopt PLP are more stable than those based on MFCCs, as can be seen from the reported values of the standard deviations over the 10 runs. The best performance is **44.67%(0.44)** which is achieved by adopting PLP and tanh as an activation function. In order to evaluate the performance of the developed systems, a comparison was made with the reported system found in the literature [50]. The results of this comparison are presented in Table 4.6, which show the superior performance of our developed system.

Activation Function	Systems	Accuracy Rate
Sigmoid	ESN& MFCCs	41.57 % (0.83)
	ESN& PLP	42.10 % (0.39)
Tanh	ESN& MFCCs	36.99 % (0.58)
	ESN& PLP	**44.67 % (0.44)**

Table 4.5: The results obtained by the four developed system, we report the mean over 10 runs and the standard deviation.

Systems	Accuracy Rate
Combined Learning (Nadia Hmad et al,2013)[50]	38.20 %
Echo State Network & PLP (This work)	**44.67 % (0.44)**

Table 4.6: The results obtained by the best ESN system and from the compared study.

4.4.3.3 *Discussion*

The results of these experiments show that the use of ESN in developing a phoneme-based speech recognition system can improve performance. This

holds true even for a limited resource language such as Arabic, where only small-scale corpora are available. The findings of these experiments also suggest that the choice of the activation function and the pre-processing approach influence the overall performance of the system. This means that, to achieve the best possible performance, these two selections needs to be optimised to suit the task at hand. The PLP approach provides better performance, regardless of the used feature extraction method which promotes the adoption of PLP in developing ESN-based systems. The proposed system architecture that combines PLP and ESN with the use of Tanh provides superior performance compared to the considered study.

The improvements achieved by changing only the activation function encourage an investigation of the effect of applying new activation functions. The nature of ESN supports the adoption of new activation functions as, unlike error backpropagation-based architecture, it does not require the derivative of the used activation function to be computed. To sum up, the findings of this study suggest that adopting ESN to develop an Arabic phoneme-based speech recognition system can improve performance and the developed system provides better performance compared to the previously-published work. This promotes a wider adoption of ESN in the Arabic speech recognition system and encourages the development of novel architectures to build upon these promising results.

4.5 CONCLUSION

In this section, we discussed the findings of all of the conducted experiments across all of the considered corpora. We also revisited the objective of this chapter, emphasising how these results contribute towards its achievement. In addition, the implications of the findings and possible new research directions

are presented. Finally, the limitations of this study are covered and potential solutions to overcome them in future work are offered.

The objective of this chapter is to investigate the potential advantages associated with developing an ESN-based system for a limited resource language, namely Arabic. In order to achieve this objective, several ESN-based speech recognition systems have been developed for Arabic and evaluated across several corpora. The findings of the conducted experiments show the superior performance of ESN-based systems which encourage a broader adoption in the field. This robust performance is seen across all of the considered corpora, which have different properties such as size, number of speakers, number of classes, labelling system (phoneme or word-based labelling) and finally the recording settings. Several methods have been used to evaluate the developed systems that include comparisons with the systems found in the literature, developing baseline models and reproducing the reported results found in the literature. It is clear, from the results of the conducted comparisons with previous work found in the literature, that ESN can improve the performance of Arabic speech recognition systems. This finding is supported by the results of comparing the developed systems with baseline models where ESN shows a better performance again. It is important to note that the compared systems (found in the literature or baseline models) have enjoyed decades of research that ESN as a relatively new approach has not. In other words, the performance of ESN is very promising especially when considering the limited amount of work that has been devoted to this approach. The training time is very competitive, the reservoir size controls the amount of time needed to train the ESN-based system, compared with other approaches, and this method is fast enough to handle real-time applications. ESN-based models were able to provide good performance even when only a very limited sized corpus was available with many outputs classes, a regime that is widely-considered to be a challenge to the state-of-the-art technology.

Several systems' architectures, that adopted different feature extractions, have been investigated in order to obtain the best possible performance. A comparison among these preprocessing approaches (namely PLP, RASTA-PLP and MFCCS) have been conducted to show the advantage of using RASTA-PLP, where possible, is that the utterance is long enough to apply the RASTA filter, or PLP in the other cases and the MFCCs approach has the poorest performance across the considered approaches. Thus, we propose two models based on these findings. The first is for word-based systems (utterances are typically long enough to apply a RASTA filter) and the other is for phoneme-based systems. The first architecture uses RASTA-PLP in the pre-processing stage and ESN for the classification stage, whereas the second architecture uses PLP in the pre-processing stage and ESN for the classification stage. These two proposed systems have shown very promising results compared to the previous systems found in the literature. Another important finding of this work is that the choice of the activation function has a significant impact on system performance and, to achieve the best performance, the feature extraction methods and activation function need to be optimised at the same time, as following a greedy approach may not lead to optimal performance.

Every research project suffers from limitations and this work is no exception. The first main limitation is the absence of publicly-available corpora that can be used to develop large vocabulary ESN-based speech recognition systems. Creating such a corpus that covers thousands of words is needed to understand how the system behaves in such regimes but such a task falls beyond our research constraints. However, we think that the results presented in this work will attract more funding to this work and we plan to start developing such a corpus when the funding becomes available. The other main limitation is also related to the corpora available for this study, as the Arabic language has many different accents that vary significantly [30], making it very challenging to develop a corpus that covers all of these differences. Finally, the limited

computational power that is available to this research prevented us from experimenting with a larger reservoir size and studying the effects of such changes.

There are several implications for the work that has been presented in this chapter. Firstly, more work is needed to investigate the novel architectures of ESN-based Arabic speech recognition systems to improve the robust performance reported in this work. This includes building different systems and different corpora for the Arabic domain. Secondly, the fundamental properties of ESN need to be studied further to produce a better understanding of the reasons behinds its robust performance. Empirical and theoretical efforts are required to achieve this goal which will allow us to improve this approach further. Investigating different read-out functions, activation functions and novel topologies are among the most promising variables that might improve the conventional ESN approach. In addition, it is important to test ESN-based systems in the presence of noise and different environments, which are largely considered to be among the main challenges facing the state-of-the-art techniques.

5

NOVEL ARCHITECTURES FOR ECHO STATE NETWORK

5.1 INTRODUCTION

In this chapter, we present several novel architectures to improve the performance of ESN-based speech recognition systems, inspired by the robust performance reported in the previous chapter. There are two main proposed architectures that focus on improving the classification capability of the read-out function. In designing these novel approaches, the emphasis has been made on maintaining the main attractive properties of the conventional ESN structure which include robust performance, fast training and convex solutions for the read-out function optimisation problem. The focus on improving the read-out function rather than other ESN components is based on evidence found in the literature[91][92]. This evidence suggests that the reservoir response maintains adequate information to classify the different classes. This has been demonstrated by developing two systems. The first uses the raw data, MFCCs, with a state-of-the-art speech recognition system while the other develops an ESN-based system then uses the reservoir response instead of the raw data to train the same state-of-the-art speech recognition system. Both systems have similar performance, which suggests that there is sufficient information in the reservoir response. In addition, the read-out function contains the only learnt parameters, which creates several challenges for conducting this training in a fast and efficient manner. This is particularly challenging when the classification task is not a binary problem, as there are more than two different classes, which is the case for the speech recognition problem.

The evaluation process includes comparing the system to the conventional ESN and other previous systems found in the literature. We discuss in detail these proposed systems and the motivation behind this design. The limitations of each design and the regimes where the systems perform best are also included in this chapter

5.2 A NOVEL APPROACH COMBINING AN ECHO STATE NETWORK WITH SUPPORT VECTOR MACHINES

In this section, we describe a novel approach for Arabic speech recognition systems based on ESN. This approach builds upon the robust performance of the conventional ESN architecture discussed in the previous chapter and aims to improve it even further. In the standard ESN approach, a large reservoir tends to be required to achieve state-of-the-art performance, mainly due to the use of a linear read-out function in the output layer. Typically, in the linear read out function setup, the input is mapped to a very high dimensional space where the different classes can be linearly separated. This is a computationally expensive and time-consuming approach that increases the danger of over-fitting the training data, which may result in degrading the generalisation of the developed system. This problem can be avoided when large datasets are available to train the system but, even then, hardware limitations can prevent the mapping of such a large corpora to a higher dimension. Thus, the aim of this work is to develop a new technique that addresses these problems, whereby a more robust performance can be obtained even when only a small to medium sized reservoir is used. To achieve this goal, we have developed a novel approach for Arabic speech recognition that combines echo state networks with support vector machines. This developed approach has been evaluated on the only publicly-accessible Arabic speech corpus, namely SAD. The results have been compared with those obtained by using a conventional

ESN system and other systems reported in the literature. We have published this technique in[2] with the results of these comparisons, and some of this published material is presented in this section. The strengths and limitations of this approach are also covered in this section, and possible improvements are discussed.

5.2.1 *Motivation*

The main motivation behind the development of this model is that the linear read-out function used in the output layer in ESN has a very limited classification ability. This means that, to achieve state-of-the-art performance, a huge reservoir needs to be used in many real-world applications to find a feature space where the different classes are linearly separable. This may be problematic as it can lead to dealing with a regime where the number of degrees of freedom is far bigger than the sample size so applying the simple linear read-out function to calculate the output weights may result in severe over-fitting. The generalisation error bounds will also be invalid in a such regime. In addition, a linear read-out is sensitive to outliers, which means that noisy data can severely affect performance. Another issue that arises from applying the simple read-out function is that it is possible to end up with a non-invertible matrix as a response to the reservoir dynamic, which leads to several issues associated with learning the final weights in the output layer.

Based on the previous argument, we suggest replacing the simple linear read-out function with SVMs (SVMs were discussed in 3.3.1.6). Adopting SVMs increases the classification capability of the output layer which allows the system to provide a better performance even when a relatively small number of reservoirs is used and the different classes are not linearly separable. In addition, it allows the system to maintain many of the attractive characteristics of the conventional ESN. This includes the convex optimisation solution in

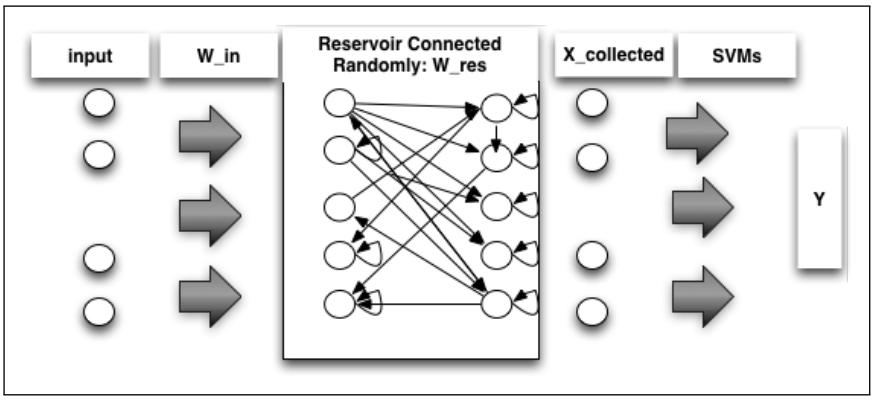

Figure 5.1: The proposed system (ESNSVMs) Structure where the linear read out function in the output layer is replaced SVM classifiers.

the training phase of the read-out function that avoids many issues raised by using non-convex cost functions such as determining the initial points, local optima and terminating criteria. All of these issues increase the difficulty of developing approaches that adopt a non-convex function and limit the results' reproducibility. In terms of generalisation, SVMs provide an error based on the number of support vectors, estimating the decision boundary. It is important to state that this developed approach can be seen as a mechanism that allows SVMs to handle temporal data. In other words, it provides the standard SVMs approach with a memory where a time dependency structure can be captured and modelled.

5.2.2 *Proposed Approach (ESN & SVMs)*

We present a detailed description of the proposed model and practical guidelines about its implementation to allow the reader to apply it to new tasks or reproduce our reported results. However, the strengths and weaknesses of this approach will be discussed in the Discussion section. The proposed approach can be described as follows:

First, mapping the input vector to higher dimensions using $\mathbf{W^{in}}$, which is a p by r matrix where p is the dimension of the input vector and r is the reservoir size, means that it can be initialised randomly, similar to ESN. Then, constructing the reservoir randomly $\mathbf{W^{res}}$, which is an r by r matrix, is applied in the same manner as in ESN and collects the response of the reservoir in $\mathbf{X^{collected}}$, which is a matrix m by r where m is the number of samples. This matrix $\mathbf{X^{collected}}$ with the target label $\vec{y}$ is used to train SVMs, which is a vector of m components where the i^{th} element represents the class label for the i^{th} sample in the training set. To predict a new data point, it is necessary to map using $\mathbf{W^{in}}$ and $\mathbf{W^{res}}$, and feed the output to the trained SVMs to determinate the class label of the sample. A summary of these steps is provided below [1]:

- Step 1: Map the input signal using $\mathbf{W^{in}}$ and pass it to the reservoir the reservoir $\mathbf{W^{res}}$ for time 0.
- Step 2: Repeat the same procedure until the end of the signal (different samples do not need to be the same length) and collect the response of the reservoir in $\mathbf{X^{collected}}$.
- Step 3: Use $\mathbf{X^{collected}}$ and target label $\vec{y}$ to train single SVM classifiers in the binary classification problem or multiple SVM classifiers in the multi-classification problem.
- Step 4: Predict a new data point by using the mapping procedures described in steps 1 and 2 and applying the learnt SVM classifiers on the response of the network to determine the label of the new sample.

To optimise the parameters of the reservoir and SVMs, we suggest using a validation set to optimise the hyperparameters of an ESN model with a small

1 Note: training is step 1,2,3; testing is 1,2,4.

reservoir size. Once this has been accomplished, the reservoir size can be increased and the output of a reservoir with the same parameters estimated in the previous step is used to select the SVMs parameters such as the kernel type and the cost function based on their performance on the validation set. These steps usually help to reduce the time needed for optimising the proposed approach, especially when dealing with multiple label classification tasks.

5.2.3 *Experiments*

To evaluate the performance of the suggested approach, a publicly-accessible corpus namely SAD (see 4.3.1 for a detailed description) has been used to conduct several experiments, which aim to explore the proposed system performance compared to the conventional ESN approach across different reservoir sizes. The results of these experiments are also compared to published approaches found in the literature that used the same corpus[41][16].

5.2.3.1 *Hyperparameters Optimisation & Implementation*

Parameter selection falls under the umbrella of the model selection phase and the techniques vary with regard to their sensitivity to changes in the hyperparameters. RNNs (apart from reservoir-based RNNs) are known to be sensitive to weight initialisation, which limits the ability to reproduce the result even when using a similar architecture. On the other hand, SVMs are more robust to changes in the initial weights and reproducibility is more likely to occur when fixing the kernel type and the regularisation parameter, mainly due to convex optimisation, which results in a global minimum solution.

The hyperparameters of each model tend to affect overall performance differently, which leads researchers to focus on the most important in terms of their impact. The typical method of selecting hyperparameters, which is also adopted in this experiment, is to use a subset of the training set, which is

known as a validation set, and test a variety of hyperparameter values. The values corresponding to the best result from the validation set will be selected. Once the hyperparameters are fixed, the model is tested on the test, unseen, dataset and the results are reported and compared with other approaches. ESN has several hyperparameters that need to be set empirically using the validation set; however, their impact on performance varies significantly. The two major hyperparameters that need to be determined are the reservoir size and the leakage rate as these both have a major impact on performance. Finding the optimal value that maximises performance may require a sound background in machine learning, as using a very large reservoir may easily result in high variance, which needs to be addressed by adopting the appropriate regularisation technique. In determining the leakage rate, prior knowledge of the nature of the task dynamic is useful. In this experiment, using a leakage rate value larger than 0.4 prevents the model from distinguishing among the Arabic digits 4, 7 and 9, as they all end with the same sound. The other hyperparameter is the input scaling constant, which is optimised to obtain the desired dynamic of the reservoir. However, in the literature it is reported that it does not affect performance severely, and the result of this experiment supports that view. The values used in ESN are: reservoir size = 900, leakage rate =0.005 and scaling constant = 1.75. In ESNSVMs, the same previous parameters and the RBF kernel are applied with gamma 0.001 and the cost value is 1,000, known as the regularisation parameter.

The SDA corpus contains 8,800 samples which are divided as follows: 6,600 instances for training and 2,200 instances for testing. The training set was divided into two parts, with one used for training, which contains almost 75% of the training sample (5,000 samples), and the other used as a validation set (containing 1,600 samples) in the model selection phase. Here, we report the result on the test data, which was not used in the development process of the model. This corpus is available only in preprocessed format, with

13 Mel-frequency cepstral coefficients (MFCCs). Matlab code was written to implement ESN and the LIBSVM library[17]. The Matlab version was applied to train SVMs classifiers in the output layer of ESNSVMs.

5.2.4 *Results*

In this section, the results of the conducted experiments will be described and compared with the previous published work. The results are summarised in table 5.1, where ESN, the proposed system (ESNSVMs) and the considered studies are compared. It is clear from these results that ESNSVMs provides the best performance compared to the other considered systems. The ESN system is also superior to the other two approaches, which is consistent with our findings stated in the previous chapter.

Systems	Accuracy Rate
TM (Nacereddine Hammami et al, 2011)[41]	94.04%
LoGID (Paulo R. Cavalin et al,2012)[16]	95.99%
Echo State Network	96.91%
Proposed System (ESNSVMs)	**97.45 %**

Table 5.1: The results obtained by the proposed system , ESN and from the two compared studies

In table 5.2, we investigate in-depth the performance of the proposed system in light of the work by Nacereddine Hammami et al [41]. The accuracy of each class is presented and compared and the average across all of these classes is computed. ESNSVMs outperforms the compared approach in almost every class and achieves an overall average accuracy of 97.45 compared to 94.04. These results show the robust performance of the proposed system and

encourage us to adopt and improve it. Unfortunately, such a comparison with the other approach, LoGID (Paulo R. Cavalin et al, 2012)[16]), is impossible, as these detailed results have not been published.

English	Arabic Sounds	Arabic	TM	ESNSVMs
Zero	.sifr	صفر	93.28	98.6
One	wahid	واحد	99.95	97.7
Two	itnan	اثنين	90.19	97.7
Three	talatah	ثلاثة	92.16	98.6
Four	arbaa	أربعة	94.59	96.3
Five	hamsah	خمسة	97.62	98.6
Six	sittah	ستة	95.35	95
Seven	saba	سبعة	89.27	93.6
Eight	tamaniyyah	ثمانية	92.98	99
Nine	tisah	تسعة	95.51	99
		Average	94.04	**97.45**

Table 5.2: The result obtained by ESNSVMs for each digits compared with TM approach

The effect of reservoir size was examined in these experiments and the results of using different reservoir sizes are compared to ESN, is shown in figure 5.2. These results indicate that the use of the suggested system improves performance over the standard ESN approach particularly when the reservoir size is small. An improvement of 15% is achieved when the smallest reservoir size (the reservoir is equal to the size of the input signal dimension) is used and this margin continues to decrease as larger reservoir sizes are used. This is mainly due to the fact that using a larger reservoir size means increasing

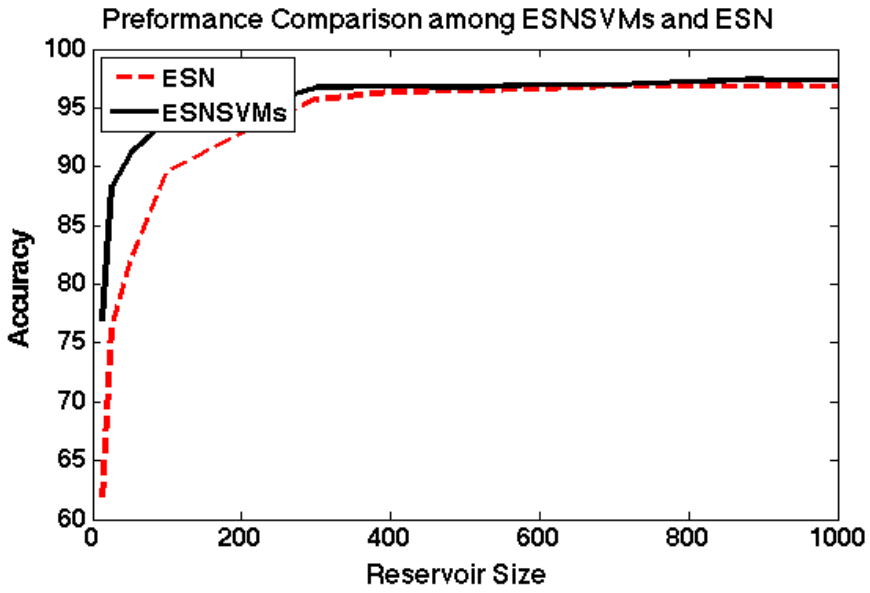

Figure 5.2: The effect of the Reservoir Size on the Performance of ESN and ESNSVMs.

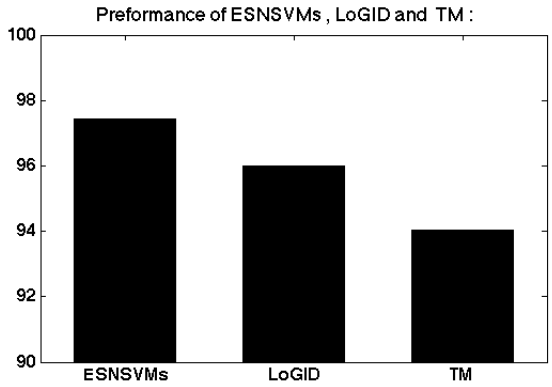

Figure 5.3: A comparison among ESNSVMs, LoGID and TM

the dimensional space of the output layer which makes it easier for the linear read-out function to find a linear decision boundary.

A confusion matrix of the best results obtained using the proposed system is presented in figure 5.4. This shows that numbers 8 and 9 have the best accuracy (99%) whereas number 7 has the lowest accuracy (93.6%). This lower accuracy may be due to the similarity between numbers 4 and 3 in pronouncing the last syllables which can explain the system's confusion when separating these classes. Other mistakes made by the systems are less obvious and a direct link to the similarity between digits' pronunciation cannot be established. These

0	1	2	3	4	5	6	7	8	9	
217	0	0	0	0	2	0	1	0	0	0
0	215	1	0	0	0	0	0	4	0	1
0	2	215	0	0	0	0	0	2	1	2
0	0	0	217	2	0	0	0	1	0	3
0	0	0	5	212	0	0	3	0	0	4
1	0	0	0	1	217	0	1	0	0	5
7	0	0	0	0	2	209	0	0	2	6
1	0	0	10	3	0	0	206	0	0	7
0	0	0	1	1	0	0	0	218	0	8
2	0	0	0	0	0	0	0	0	218	9

Figure 5.4: Confusion matrix of best result obtained by ESNSVMs

mistakes can be the result of a loss of information during the pre-processing stage or different environmental aspects that affect system performance.

In summary, the developed systems have been proven to be superior to both the conventional ESN approach and other state-of-the-art published systems. The performance of ESN and ESNSVMs is heavily affected by the reservoir size. Where a small reservoir is used, the developed systems outperforms ESN by a significant margin. In analysing the mistakes made by the systems, many classification errors can be explained as a similarity in pronunciation that confuses the system.

5.2.5 *Discussion*

Based on the previous obtained results, we argue that using ESNSVMs can improve system performance. Moreover, it is clear from our experiments that, when using a small to medium reservoir size, ESNSVMs achieves a significant improvement in the accuracy rate. This might be particularly attractive when dealing with time series with a very high dimension e.g. image sequences. Also, the ESNSVMs shows robust performance against over-fitting, with easily computed error bounds of the SVMs, which offers an estimation of the model's generalisation. Developing new kernels to tackle a specific problem

is also enabled when using ESNSVMs, which may lead to improvements in performance for different tasks.

The suggested model has some limitations that could prevent it from achieving the desired performance in certain regimes. The main limitation of ESNSVMs is the added complexity over ESN, which is represented by the need to optimise the SVMs parameters used in the output layer. This includes the choice of kernel and its associated parameters. The learning in general will take longer, especially when applying ESNSVMs to multiple class problems with medium to large numbers of classes. This is due to the nature of SVMs, which are binary classifiers that require constructing at least equal to the number of problem classes, in the one against all approach, or $n(n-1)/2$ classifiers where n is the number of classes, in the one against one approach.

This limits the ability of ESNSVMs when dealing with the multi-class classification problem with a medium to large number of classes which is required in developing phoneme-based speech recognition systems. In the Arabic domain, using ESNSVMs to develop a phoneme speech recognition system means that 1,260 classifiers needs to constructed in the output layer. This is a computationally expensive and time-consuming procedure that may prove impossible due to hardware limitations. Thus, this developed system has an advantage over the standard ESN approach in regimes where the input dimension is input dimension is very high and there is a limited number of classes.

Another issue with this developed approach that both ESN and ESNSVMs suffer from is that they both rely on the offline learning paradigm. This can be seen as an advantage to these approaches when the available data are relatively manageable and can fit on a single machine, but when a large dataset is employed, these two approaches cannot utilise this valuable resource. This is a critical property that is required in almost all tasks nowadays as an increasing amount of data becomes available to train classification systems.

An approach that can be distributed over several machines or implemented in an online learning paradigm is more likely to be able to take advantage of a significant increase in data volume.

In addition, the inability to handle unlabelled data, which are the most available data compared to labelled data, is a related limitation that stems from excluding any unsupervised learning elements in developing these two approaches. This is not only due to the available volume of unlabelled data but there is also a financial factor, as unlabelled data are considerably cheaper than labelled data. This will be reflected in the overall cost of developing the required systems. This prompts the investigation of new techniques to incorporate the knowledge of the unlabelled data to improve the robustness of ESN and ESNSVMs and reduce the development costs of systems based on these two approaches.

Future work will include investigating other possible classifiers to avoid the limitation of SVMs in dealing with multi-class tasks with a large number of categories. This aims to develop an approach that can offer the same improvement in performance presented as here in these experiments to phoneme-based speech recognition systems. In section 5.3, we present a novel approach that addresses this issue and provides a practical solution that offers even greater improvements in performance while also being able to handle tasks with a large number of classes.

5.2.6 *Conclusion*

We have proposed the ESNSVMs approach, which combines ESN and SVMs for Arabic speech recognition. To evaluate the performance of the ESNSVMs, we conducted an experiment employing the well-known Arabic spoken digits. The dataset is publicly available and contains 8,800 samples of the Arabic digits 0-9. The result has been compared with ESN and two other state-of-

the-art models. ESNSVMs achieves a high accuracy rate of 97.45%, which demonstrates the potential of applying it in Arabic speech recognition systems with a small number of classes. In addition, the proposed approach achieves a higher accuracy rate than ESN, especially when the reservoir size is small. Further work will include reducing the model complexity of the output layer to give ESNSVMs the ability to handle multi-class classification problems with a large numbers of classes.

5.3 NOVEL APPROACH COMBINING ECHO STATE NETWORK WITH EXTREME KERNEL MACHINES

In this section, we propose another novel approach that takes into consideration the limitations of the previously-suggested systems, described in the last section. This approach aims also to improve the performance of the conventional ESN approach by increasing the classification capability of the read-out layer. In order to achieve this aim, the simple linear read-out function is replaced by the extreme learning machines (ELM)-based approach, namely extreme kernel machines. This new approach has a signification advantage over the ESN and ESNSVMs which has been demonstrated by conducting several experiments across different corpora. We start by motivating our novel approach in the motivation sections and then describe the model in-depth and compare it with ESN and ESNSVMs in the proposed system section. The experiments are discussed individually in the experiments section, then we discuss the findings of all of these experiments in the discussion section. Finally, we draw our conclusions in the conclusions section.

5.3.1 *Motivation*

Based on our previous work on developing ESNSVMs, it is clear that adopting a nonlinear readout function can improve the performance of the conventional ESN approach. This has been demonstrated by the superior performance of the proposed ESNSVMs approach; however, the limitation of this approach in handling multi-classes tasks with a large number of classes imposes a serous barrier to its wider adoption for many speech recognition tasks. This has motivated us to develop a novel approach that inherits the robust performance of ESNSVMs and is also practical for multi-classes tasks.

The motivation behind the selection of EKMs over other nonlinear state-of-the-art learners' technologies is based on several elements. Firstly, in designing this new approach, we wanted to retain all of the attractive properties of the ESN approach. This meant that the output layer must be learnt quickly and the results must be easily reproducible, so a single-shot solution for a convex cost function is needed. These desired properties are unavailable from other state-of-the-art classification approaches, such as feedforward multilayer perceptrons, where the cost function is highly non-convex. On the other hand, EKMs provide the required increase in classification power of the output layer while maintaining a convex cost function (see section 3.3.3.5 for more details). In addition, EKMs cannot handle a temporal structure so we therefore combine it with ESN to provide the conventional EKMs approach with a mechanism to overcome this limitation. Thus, we have designed the structure of the proposed approach to take advantage of the elements of strengths of each approach individually and combine them to overcome the described limitations of each approach.

5.3.2 *Proposed Approach*

In this section, we describe the proposed approach in detail and provide an implementation guide to extend it to new tasks. The proposed model aims to improve the classification capability of the ESN by applying the EKM classifier on the output layer instead of the linear classifier used in the conventional approach. We found few attempts in the literature [90][2] to overcome the limitation of the linear readout function and replace it with a nonlinear classifier. The added training time (or the binary nature of some classifiers such as SVMs) makes this impractical for many tasks, specifically for multi-label tasks with a relatively large number of classes. However, using EKMs yields the benefits of using the nonlinear function while maintaining a single-shot convex solution that can handle multi-label tasks even when the number of labels is large. This is important not only from the efficiency aspect but also from reproducibility: many nonlinear classifiers such as multilayer perceptrons are sensitive to their initial weights. In addition, we have found that dividing the signal into subparts and separately feeding these subparts to the reservoir improves performance.

The proposed model is implemented in three main stages. The first stage is to develop a conventional ESN system to perform the task at hand. A small reservoir size is typically used to implement this system to reduce the time needed to optimise the systems. Once the reservoir's components are optimised, (namely $\mathbf{W^{in}}$, $\mathbf{W^{res}}$, the leakage rate and the activation function) the second stage is started by running the reservoir using a larger number of nodes(typically, the largest possible reservoir size is used). The third stage consists of using the collected reservoir's responses to train the EKM classifier used in the output layer (instead of the simple linear readout function used in the standard ESN approach). This includes optimising the kernel type and all of its related parameters and the regularisation coefficient that control the

complexity of the model and are used to avoid over-fitting. An implementation step by step guide is summarised as follows:

- Divide the corpus into three different subsets (training, validation and testing).
- Optimise a small reservoir network using the standard ESN (only training and validation subsets are used).
- Run the data through a larger reservoir size and collect the reservoir responses. The size of the reservoir is determined by the computational power available, using training and validation subsets.
- Use the collected responses to optimise EKMs classifier, which is used in the output layer.
- Combine the training, validation subsets in a single set and train the model on it.
- Evaluate the performance of the developed system on the test, unseen, subset which has not been used in any of the previous steps.

This practical guide allows the rapid, efficient implementation of the proposed approach. The novel architecture suggested here by adopting EKMs increases the classification capability of the output layer. A kernel is used in the output layer to map the reservoir responses to a new, typically higher, dimensional space where a linear classifier is applied. This means that the linear classifier is applied in a space determined by the training sample size (the new reservoir response representation is an m by m matrix where m is the number of samples), so the sparse solution offered by the ESNSVMs is lost. If the linear kernel is selected, the model complexity can be significantly reduced; however, a nonlinear kernel such as RBF tends to offer richer mapping which is reflected by achieving higher accuracy where the different classes

cannot be linearly separated. It is important to emphasise that the selection of the kernel type is part of the model selection phase which is handled as a hyper-parameter optimisation problem and the validation set is used to find the most effective kernel for the task at hand. The use of a simple, linear, kernel reduces the overall model complexity but may degrade the accuracy so a tradeoff needs to be considered.

Although this proposed model increases the complexity of the output layer, the overall system complexity required to handle the considered task is typically reduced. The results we present in the next sections show that the developed system facilitates a significant improvement in performance that ESN cannot achieve, even when the reservoir size is increased by 100%. This also shows the robustness of the suggested approach compared to the standard approach. The limitations and strengths are discussed further after presenting the results of the conducted experiments in the discussion sections.

5.3.3 *Experiments*

In this section, we present the experiments conducted to evaluate the proposed approach. These experiments have been organised based on the corpus employed, so that each subsection covers the experiments conducted on a specific corpus. In each of these subsections, there is a detailed description of the conducted experiments with their results and a discussion of their findings. The findings of all these experiments are discussed in the discussion subsection which covers the results of all of the conducted experiments across all of the corpora to present the complete outcome of the evaluation of the suggested approach. This includes the strengths and limitations of this novel approach. Finally, we draw our conclusion in the conclusion section.

5.3.3.1 *Experiments on the Spoken Arabic Digit Corpus*

One of the best practices in evaluating novel systems is applying the approach to public corpora. This allows developers to conduct direct, valid comparisons between the evaluated system and other state-of-the-art approaches. It also ensures the reproducibility of the reported results which is a crucial element in conducting research. Thus, we use the only publicly-accessible spoken Arabic digit corpus (SAD) to evaluate our system (see Chapter four for a detailed description of SAD). This corpus is only available in a preprocessed format (13 MFCCs coefficients), so using different feature extraction techniques or the addition of noise is impossible with this corpus. The results obtained are compared with the standard ESN baseline model and other state-of-the-art approaches, reported in the literature.

5.3.3.1.1 HYPERPARAMETERS OPTIMISATION & IMPLEMENTATION

The hyperparameters of the ESN system have been optimised following the same procedures described in the previous chapter. The training data are divided into two subsets (training and validation sets) which are both used to optimise the model hyperparameters using a small reservoir size. Once this has been accomplished, a larger reservoir size is used and the regularisation coefficient is optimised to avoid any over-fitting-related problems. In this experiment, we were able to construct a model with a considerably larger reservoir than the demonstrated models in the previous chapter, as more computational power was available. Finally, the system is trained on the two subsets (training and validation), and then evaluated using the test set, which has not been used previously. We ran the test 10 times and reported the mean and standard deviation. Similar steps have been followed in optimising the novel proposed model (ESNEKMs). The reservoir response of the optimised ESN model has been used to optimise the EKMs classifier on the output layer. A

linear, polynomial and Radial basis function (RBF) kernel has been considered in the model selection phase and the RBF has been selected based on its performance on the validation sets. Once the EKMs classifier was optimised, we used the two subsets (training and validation) to train the classifier and evaluate the system using the test set. Like ESN, we ran the test 10 times and reported the mean and standard deviation. MATLAB code has been written to implement the ESN approach and the library [58] is used to construct the EKMs classifier used on the output layer.

5.3.3.1.2 RESULTS

The first thing to notice here is the improvement in the performance of the standard ESN and ESNKMs compared to our work described in the previous chapter and published in [2](see table 5.3). This is due to the fact that we were able to construct a larger reservoir, as the computational power available to construct the system has increased. This supports our conclusion based on the previously-conducted experiments, where we claimed that increasing the reservoir size can improve system performance. The results of the two systems shows superior performance compared to the three considered state-of-the-art approaches reported in the literature. This demonstrated the potential of adopting ESN to develop Arabic speech recognition systems.

The results of ESN and ESNEKMs are close to each other; however, the proposed approach offers a more stable performance, as can be seen by comparing the values of the standard deviation. ESNEKMs provides relatively higher performance and a significant reduction in the standard deviation (almost 50% lower than that for ESN).

The effect of reservoir size on performance has been examined and compared using the conventional ESN and ESNSVMs (see figure 5.5). The results show that the proposed approach (ESNEKMs) outperforms the other two approaches even when a small reservoir size is used. In fact, the smaller the reservoir size,

System	Result
TM (Nacereddine Hammami et al, 2010) [41]	93.10%
CHMM (Nacereddine Hammami et al 2012) [43]	94.09%
LoGID (Paulo R. Cavalin et al,2012)[16]	95.99%
ESN(This work)	99.06% (0.23)
ESNEKM(This work)	99.16% (0.12)

Table 5.3: The results obtained by the proposed system , ESN and from the compared studies.

the higher the increase in accuracy. The largest margins are 20% and 4.8% compared to ESN and ESNSVMs respectively, achieved when the smallest reservoir is used, and the reservoir size is the same as the input size, which is 13. The improvement in performance continue even when a larger reservoir size is used (increasing the reservoir size by a factor of 2 for every experiment) and the system reached a 98% accuracy rate when only a reservoir containing 100 nodes was used which other systems cannot obtain even with a larger reservoir. The other system cannot match the performance even when the reservoir size is increased 10-fold, to a 1000 nodes. This shows the robust performance offered by this novel approach with a small reservoir, which means that a far less complex model can be constructed compared to ESN and ESNSVMs. This is reflected in the time and computational power needed to train the model and test it using a large reservoir, as the model's parameters increase as the square of the reservoir size.

In summary, the developed system provides a significant improvement in performance compared with the other state-of-the-art techniques reported in the literature and two baseline models (ESN and ESNSVMs). This improvement

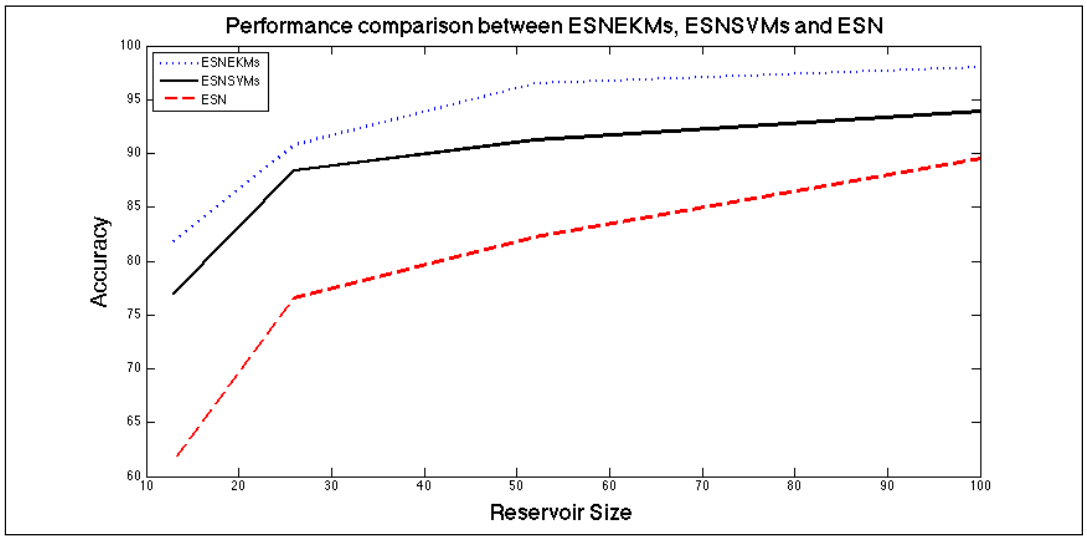

Figure 5.5: The effect of the Reservoir Size on the Performance of ESNEKMs, ESNSVMs and ESN.

is achieved across different reservoir sizes, from 13 to 1,500 nodes, although the margin tends to decrease as a larger reservoir is used. A robust performance using a small reservoir is offered by this approach that ESN and ESNSVMs cannot match even when applied to a far larger reservoir size (increasing from to 1,000 nodes). In addition, ESNEKMs provides more stable performance, as shown in the decrease in the standard deviation over 10 runs.

5.3.3.1.3 DISCUSSIONS

Based on the previously-described results, the proposed approach provides superior performance across different regimes compared to the state-of-the-art approaches. Evaluating the performance using different reservoir sizes provides an insight into the robustness of each model. The main findings of these experiments can be summarised as follow:

- The proposed system offers superior performance to all other considered systems (ESN, ESNSVMs and the models reported in the literature).

- There is a significant margin in performance when a small reservoir is used that the other models cannot eliminate even when the reservoir size is increased by a factor of 10.
- A more stable performance is obtained by the ESNSKMs compared to the considered techniques.
- The gap in performance tends to decrease as a larger reservoir is used.

All of these findings demonstrate the potential of ESNEKMs and promote the wider adoption of the proposed approach as it overcomes many of the limitations of ESN and ESNSVMs. The need for a large reservoir size, which means more complex models that are computationally expensive to train and run, and the inability to handle multiclass tasks with a large number of classes in the case of ESNSVMs are addressed in this suggested model. This is combined with a superior, more stable performance. However, this robust performance stems from the powerful classification techniques used in the output layer, namely EKMs, which needs to be optimised to achieve the desired performance. This means that there is an added complexity to the read-out function compared to ESN and the sparse solution offered by ESNSVMs is lost. Despite this, the increase in complexity in ESNSKMs is less problematic as the model provides significantly higher accuracy compared to ESN and ESNSVMs with a smaller reservoir size which means that ESNSKMs can outperform the other two approaches with significantly fewer parameters, thus significantly reducing the required complexity. A major limitation of this approach is that the model cannot utilise unlabelled data; however, this limitation is present in ESN and ESNSVMs as well. In addition, the proposed model has not been evaluated using noisy data, which is more reflective of the approach's performance regarding real-world tasks, as adding noise to this corpus is impossible since it is only available in a preprocessed format.

5.3.3.2 *Experiments on the Arabic Speech Corpus for Isolated Words*

In this section, we describe in detail the experiments conducted on the Arabic Speech Corpus for Isolated Words, that has been developed here (see Chapter four for more information). In designing the experiments on this corpus, many of the constraints imposed by the use of the SAD corpus have been overcome. The main constraint of using MFCCs as the feature extraction technique is no longer an issue, as this corpus is available in raw format. This is one of major objectives in developing the Arabic Speech Corpus for Isolated Words. In order to take advantage of the properties of this corpus, several feature extraction methods have been evaluated, namely RASTA-PLP, PLP and MFCCs. In the classification process, the proposed system ESNEKM is used and two other baseline models, ESN and HMM. This study goes beyond testing the model in noise free environments to include noisy environments, which is a more challenging task as many state-of-the-art approaches have serious performance issues. Developing a technique that offers robust performance on noisy data is crucial for real-world applications where a high level of noise and different environments are present. Two types of noise have been considered (babble noise and white noise) under different signal to noise ratio (SNR) levels. The aim of these experiments is to investigate the performance of the proposed system under different feature extraction techniques, noise types and noise levels, and compare it with the two baseline models.

5.3.3.2.1 HYPERPARAMETERS OPTIMISATION & IMPLEMENTATION

A total of 9 models have been developed to test all of the possible configurations regarding the preprocessing approach and the classification techniques. The same procedures described in the previous experiments for the training set have been divided into two subsets (one for validation and the other for training). These two subsets have been used to optimise all nine models' hy-

perparameters. The performance using the validation set is used to select the optimal values of the hyperparameters of each model. The optimisation has been carried out on the clean dataset only, meaning that none of these models have been exposed to noisy data during either the optimisation or training phases. In the ESN and ESNEKM, we suggest fixing the size of the reservoir to a relatively small number, while optimising the rest of parameters reduces the time needed to select the hyperparameters. A variety of feature extraction methods have been considered, namely MFCCs, perceptual linear prediction (PLP) and RASTA-perceptual linear prediction. In addition, white noise and babble noise have been added to the test set for three levels of noise: 30 db, 20 db, and 10 db. These models have been implemented in Matlab and several libraries are adopted that compute the feature extraction methods, HMMs and EKMs classifiers[66][58].

5.3.3.2.2 RESULTS

In this section, we present the results obtained from the nine developed models and compare their performance. The strengths and limitations detected will be discussed in detail in the next section, the Discussions section. All of the results are summarised in table 5.4, which includes the results across the different feature extraction methods, different classification techniques, and the different noise types and levels. The experiments conducted on reservoir-based systems are repeated 10 times and the means and standard deviations are reported to take into account the stochastic element in the random construction of the reservoir. The same reservoir size is used on ESN and ESNEKMs models. The results obtained can be interpreted from different perspectives based upon the main criteria selected. Although we emphasise and use the complete pipeline architecture as the main criteria to be compared, combining the feature extraction method and the classification techniques used in the output

layer, we consider the other criteria, as well such as performance based on the feature extraction and performance based on the classification techniques.

It is clear from the summarised results that the pipeline architecture that combines ESNEKMs and RASTA-PLP offers the best performance. This superior performance is obtained across all noise levels and types, and there is no major difference in performance based on noise type. In fact, the improvement in performance increases as the noise level increases and the performance of the other architectures starts to degrade. The pipeline that combines ESNEKMs and MFCCs provides the second highest performance among the considered nine models but its accuracy drops significantly when a high level of noise is introduced to the test data, so babble noise seems to be more challenging to this model. The lowest accuracy level is obtained by the model that combines HMM and MFCCs on the clean test and test set with babble noise added, whereas testing the model that combines HMM with PLP achieves the lowest performance reported across all of these experiments, which is 12.06%.

The performance based on the classification technique used varies significantly. Models that adopt the reservoir computing approach (ESN and ESNEKMs) provide better performance than those that uses HMM, across all of the feature extraction methods. The ESN and ESNEKMs models also show a noise robust performance compared to the HMM models. Among ESN and ESNEKMs, the latest approach offers a higher accuracy rate and the improvement varies depending upon the feature extraction method used. However, ESNEKMs is proven to be far more reliable, as can be seen from the reported standard deviations values where the ESNEKMs-based models achieve lower values than the ESN models across all of the feature extraction methods, noise types and noise levels.

The selected feature extraction method also influences the model's performance. This impact is clear on systems that adopt the RASTA-PLP during the preprocessing stage, where the best accuracy levels are reported. These

Dataset		Feature Extraction	HMM	ESN	ESNEKM
Clean		MFCCs	97.65%	98.97% (0.15)	99.59% (0.05)
		PLP	98.45%	99.16%(0.11)	99.31 %(0.09)
		RASTA-PLP	98.8 %	99.38%(0.11)	99.69% (0.06)
White Noise	30 db	MFCCs	96.4 %	98.03%(0.21)	99.05%(0.13)
		PLP	91.3 %	90.13%(0.36)	97.59%(0.17)
		RASTA-PLP	98.1 %	99.04%(0.11)	99.59 %(0.06)
	20 db	MFCCs	85.29 %	94.914 %(0.37)	94.82 % (0.30)
		PLP	51.13 %	56.07 %(6.66)	75.39 % (0.97)
		RASTA-PLP	96.05 %	97.32 % (0.33)	98.41 %(0.07)
	10 db	MFCCs	45.67 %	77.19 %(2.12)	79.50 % (0.85)
		PLP	12.06 %	19.83 %(3.83)	35.35 %(1.96)
		RASTA-PLP	81.99 %	87.48 %(1.47)	90.29 %(0.53)
Babble Noise	30 db	MFCCs	95.85 %	97.23 %(0.29)	99.35 %(0.18)
		PLP	97.05 %	97.87 %(0.36)	99.02 %(0.06)
		RASTA-PLP	98.65 %	99.22 %(0.19)	99.65 %(0.06)
	20 db	MFCCs	78.49 %	89.72 %(0.87)	94.41 %(0.34)
		PLP	86.64 %	89.47 % (2.43)	96.64 % (0.22)
		RASTA-PLP	96.75 %	97.18 % (0.42)	98.30 %(0.14)
	10 db	MFCCs	31.77 %	64.12 % (2.31)	65.48 %(0.86)
		PLP	54.23 %	56.23 % (4.82)	81.23 %(0.32)
		RASTA-PLP	85.14 %	85.45 % (8.6)	90.76 % (0.44)

Table 5.4: The results obtained by the proposed system , ESN and a baseline hidden Markov model (HMM).

levels of accuracy are achieved across all of the classification techniques, noise types and noise levels, whereas the performance of the other preprocessing approaches decreases significantly in the presence of noise. Another important observation is that the interaction between the feature extraction methods and the classification techniques varies. This can be seen from the difference in performance across these architectures, where MFCCs performs poorly when HMMs or ESN are used in the classification stages and provides the second best performance on EKMs-based classifiers. This emphasises the importance of the pipeline design as the performance of feature extractions under different classification approaches should not be the only measure for predicting the potential of this combination.

Statistical analysis has been performed on the results shown in table 5.4. A non-parametric statistical test, the Friedman test, was selected as suggested in [24] because it is more appropriate to classifiers comparison tasks. Non-parametric tests are safer than parametric tests because they do not make any assumption about the distribution of the data. The test showed that the difference in the performance among these three approaches is statistically significant with a 1% significance level. The Friedman test was then followed by a Tukey post-hoc analysis to see which approach would provide a statistically different performance from the others (pairwise comparison). The post-hoc analysis showed that all three techniques are statistically different. This difference means that the improvements in the performance achieved with ESN and ESNSVM are statistically significant, showing the potential of the developed approaches ESN and ESNSVMs.

In summary, in the absence of noise, ESN and ESNEKMs were examined under all of the considered feature extraction approaches, and approximately all of them provide similar performance. This result does not hold true for the noisy sets, as PLP-RASTA provides superior performance regardless of the type or level of noise. As in the previous experiment, ESNEKMs provides far

more stable results and outperforms the baseline model in all of the sets. The best results for both the clean and noisy sets are achieved when combining PLP-RASTA with ESNEKM.

5.3.3.2.3 DISCUSSION

In our experiments, ESNEKMs provides better, more reliable performance compared to ESN, even if the reservoir size is relatively small (100 nodes), which is consistent with our findings regarding the SAD corpus, outlined in the previous section. This is mainly due to the increase in classification capability of the readout function where EKMs is used which allows the systems to draw highly non-linear decision boundaries that the simple read-out function promoted in the conventional ESN design cannot achieve. This is reflected by the system's ability to reduce the variation in performance that may arise from the small changes caused by the random initialisation during each run. However, there are several limitations related to the proposed system, and the added complexity in the output layer of the network (resulting from replacing the linear readout function with a nonlinear function) can be seen as the major issue. This issue includes the selection of the kernel and optimising its parameters. In addition, the testing time depends on the training size. Unlike support vector machines (SVMs), the solution in EKMs is not sparse.

Based on the results reported in the previous section, it is clear that the selected feature extraction method has an influence on system performance. This and the fact that classification techniques react differently to the adoption of the same feature extraction method increases the need to devote more time to designing the overall structure of the systems rather than relying solely on the performance of the feature extraction combined with other techniques as the only indicator in proposing novel pipeline architecture.

The performance of all of the developed models has been degraded, as expected, when noise is added to the test sets. However, there are significant

differences in the decrease in performance, mainly dominated by the feature extraction method used. Models that adopt RASTA-PLP achieve the best performance when combining this with the ESNEKMs approach and maintain a competitive accuracy across all of the classification approaches used (HMM and ESN), noise levels and noise types. The lowest accuracy reported for the RASTA-PLP-based approach is 81.99% compared to 31.77% and 12.06% for MFCCs and PLP respectively. The classification approach employed also influences performance but its impact is more limited compared to the feature extraction method. This can be seen from the fact that the lowest performance is achieved by the classification approach. Regardless of the feature extraction method, ESNEKMs achieves 35.35% (1.96) compared to 19.83% (3.83) and 12.06% for ESN and HMMs. An almost 80% increase in performance can be achieved considering only the classification approach compared to an almost 300% increase in performance when considering feature extraction methods, regardless of the classification techniques employed. This shows the relative importance of pipelines components and promotes following a comprehensive approach in designing systems that focus on the complete pipelines rather than just specific aspects and search for the ideal configuration for the whole system.

To sum up, a novel speech recognition model based on RC and EKMs which we call ESNEKMs was proposed, and evaluated using a newly-developed corpus. The different feature extraction methods considered in this study include mel-frequency cepstral coefficients (MFCCs), perceptual linear prediction (PLP) and RASTA-perceptual linear prediction. The result was compared with baseline models that are based on hidden Markov (HMMs) and ESN, so that nine models were compared in total. These models were trained on clean data and then tested on unseen data with different levels and types of noise. The ESNEKM models outperformed the HMMs models under all of the

feature extraction methods, noise levels, and noise types. The best performance was obtained by the model that combined RASTA-PLP with ESNEKMs.

5.3.3.3 *Experiments on the Arabic Phonemes Corpus*

In this section, we describe in detail the experiments conducted on the Arabic phonemes corpus to evaluate the proposed approach (ESNEKMs). This corpus has different properties such as the large number of classes (33 Arabic phonemes). This large number of classes and the fact that phonemes uttered over a very short period of time pose a serious challenge to speech recognition systems. In addition, the small corpus size and larger number of classes mean that the samples per class are very limited (115, on average). Thus, evaluating the system on this corpus allows us to investigate the performance of the proposed system in a very different regime.

ESNEKMs has been tested on this corpus and the obtained results have been compared to the ESN baseline models and the only previous works published on this corpus. Two feature extraction methods have been considered in developing the models, namely PLP and MFCCs, as the segmented phonemes are too short to adopt RASTA-PLP. In addition, two activation functions have been investigated, Tanh and logistic. This means that 8 models have been developed in total and a comparison made between them and the previously-reported approaches found in the literature.

5.3.3.3.1 HYPERPARAMETERS OPTIMISATION & IMPLEMENTATION

To ensure a valid comparison. the same training and testing sets that have been used in the previous work are also used here. A validation subset containing 474 samples is extracted from the training set and the remaining 1,420 are used for training. These two subsets have been used to optimise the models' hyperparameters. We have maintained the same size of reservoir used in [50] to ensure a valid comparison between the developed systems. A grid

search has been conducted to optimise the reservoir hyperparameters on a small number of nodes. The same values as obtained in the previous step are used as input to train EKMs classifiers, also using a grid search to optimise the kernel hyperparameters (the RBF kernel is selected based on our previous experiments presented in the last section). This has been carried out separately for each feature extraction and activation function. Once the hyperparameters of these model had been optimised, we ran the model on the test, unseen set that contains 1,098 sample 10 times and report the means and the standard deviation for each model. Matlab code has been written to implement the ESN and ESNEKMs systems that use this library [27] in the preprocessing phase and this package [58] to train the EKMs classifier in the output layer for the ESNEKMs.

5.3.3.3.2 RESULTS

The results obtained using all eight developed models are summarised in table 5.5. The ESNEKMs models outperform the conventional ESN approach under all of the considered feature extraction methods and the activation functions. The best performance is obtained by the model that combines ESNEKMs with PLP and used the tanh activation function in the reservoir. It is clear from the results that both the activation function and the feature extraction method have an impact on model performance. Models react differently to changes in the activation function even when the same feature extraction method is used. The accuracy improves when Tanh is applied as the activation function rather than the logistical function on the models that use PLP. However, the opposite result is obtained on models that use the PLP technique in the preprocessing stage. ESNEKMs provides a more reliable performance as indicated by the small values of the standard deviations over the 10 runs under all of the considered activation function and feature extractions methods.

System	Activation function	Feature Extraction	Accuracy Rate
ESNEKMs	Sigmoid	MFCCs	42.15% (0.22)
ESN	Sigmoid	MFCCs	41.57% (0.83)
ESNEKMs	Sigmoid	PLP	44.15% (0.18)
ESN	Sigmoid	PLP	42.10% (0.39)
ESNEKS	Tanh	MFCCs	40.85% (0.26)
ESN	Tanh	MFCCs	36.99% (0.58)
ESNEKS	Tanh	PLP	46.27% (0.22)
ESN	Tanh	PLP	44.76%(0.44) %

Table 5.5: The results obtained by the eights developed systems, we report the mean over 10 runs and the standard deviation.

A comparison of the best-developed models based on ESN, ESNEKMs and the previous published work is shown in table 5.5. ESNEKMs outperforms the published work by an 8% margin. Recall that we employ the same reservoir size as used in that study and the same train and test sets. ESN when combined with PLP can also outperform the Nadia Hmad et al.'s 2013 approach by about 4% but the standard deviations value is double that of the ESNEKMs, which is consistent with our findings from previous experiments, as the output layer in the ESNEKMs is far more capable of overcoming any variation in performance due to changes in the reservoir response due to the chaotic element in constructing the reservoir in each run. This comparison demonstrates the robust performance of ESNEKMs and its ability to provide superior generalisation even where only a small number of samples are available to train the system. It also shows the approach's ability to handle a regime where the number

of classes is relatively large and maintains the same reliable performance reported in our previous experiments.

Systems	Accuracy Rate
Combined Learning (Nadia Hmad et al,2013)[50]	38.20 %
ESNEKM & PLP (This work)	**46.27% (0.22)**
ESN & PLP (This work)	44.67% (0.44)

Table 5.6: The results obtained by the best ESNEKM, ESN systems and the compared study.

5.3.3.3.3 DISCUSSION

Evaluating ESNEKMs on the Arabic Phonemes Corpus allows us to investigate its performance on a very challenging regime. The small sample size and large number of classes pose serious challenges in training the system. In addition, phonemes are uttered in a very short period of time compared to words, which means that the reservoir dynamic needs to capture the behaviour of short audio segments. All of these corpus properties permit us to test the proposed approach in different aspects which cannot be achieved using the previous corpus. In other words, evaluating ESNEKMs on this corpus provides new information about its behaviour as this corpus has different properties compared to the other considered corpora. The results clearly show the robust performance of the proposed approach (ESNEKMs) with regard to this challenging corpus. ESNEKMs outperforms the previously-published work and the baseline ESN model and also provides a more reliable performance. The results also indicate the impact of the activation function and feature extraction method on model performance. The selection of the tanh function as an activation function improves the performance of models that

use PLP in the feature extraction method whereas it degrades the performance of models that adopted the MFCCs technique in the preprocessing stage. This promotes considering such interaction between the feature extraction method and the activation functions in developing speech recognition systems.

On the other hand, the proposed approach has some limitations that can prevent it from achieving the desired performance in certain regimes. The main limitation of ESNEKMs is scalability, as EKMs maps the reservoir response to another dimensional space, which is often higher but not necessarily so. This mapping is controlled by the sample size which means that, in a corpus that contains a million samples, the new dimensional space has dimensions of a million by a million, so handling such a large matrix on a single machine is often impossible due to hardware limitations. There are several solutions to this that need to be investigated, such as implementing this approach in a distributed computing fashion that makes it easier to utilise several machines instead of a single machine. The other possible solution is to develop an online learning version of the proposed approach that processes the corpus sample by sample, which allows the technique to scale up to cover a very large corpus. Finally, it is possible to use a subset from the corpus instead of using all of the samples to map the reservoir response to the desired dimensional space. Another limitation of the model that is shared with ESN is the lack of utilisation of unlabelled data. The ability to learn in unsupervised mode means that huge and cheap resources will be available instantly to train systems.

In summary, ESNEKMs provides a superior, more stable performance compared to the considered systems. The selection of the feature extraction method and the activation function influence the model performance and the interaction between them needs to be considered in order to achieve the desired performance. Finally, the ESNEKMs' limitations have been discussed, which arise from its lack of ability to scale up to a huge corpora (millions of samples) and the fact that this approach cannot make use of unlabelled data. Potential

solutions have been stated and will be further discussed in the next chapter (the discussion chapter).

5.3.4 *Conclusions*

In this section, we draw our conclusion based on the results of all of the conduced experiments on all of the considered corpora. In previous sections, we discussed the findings of the experiments based on the corpus used but here we focus on the big picture and emphasis the overall findings of the experiments. The main finding, that is consistent across all of the conducted experiments on all of the adopted corpora, is that the proposed approach (ESNEKMs) outperforms the conventional ESN technique. The reported improvements in performance increase when a small reservoir size is used. The fact that the proposed approach provides a more reliable performance has been observed across the considered corpora. ESNEKMs has been proven to be able to handle multiclass tasks with a relatively large number of classes. A superior generalisation is obtained by ESNEKMs compared to ESN, even when only a small number of samples is available to train the system.

ESNEKMs has outperformed our developed approach (ESNSVMs), which is described in 5.2. It provides a higher accuracy even when a small number of reservoirs is used. In fact, we have shown that, in some cases, ESNEKMs can provide a performance level that ESN and ESNSVMs cannot match even when the reservoir size is increased by 10 fold, ESNEKMs, with a 100 node reservoir size, outperforms ESN and ESNSVMs, that have reservoirs of 1,000 nodes. The main limitation of ESNSVMs in handling multiclass tasks is avoided in ESNEKMs, as it has the ability to handle multiclass tasks even with a relatively large number of classes.

The role of the activation function has also been investigated in this work. It is clear from our findings that the activation function influences system

performance. However, the effect of the selection of the activation function can be positive or negative based on the feature extraction employed. This encourages us to recommend optimising the type of activation function and the feature extraction method in a combined process as a greedy optimisation process can fail to capture this interaction between these two model elements.

The findings also suggest that feature extraction methods have a significant impact on performance. Thus, efforts should be made during the design phase to construct the best possible pipeline architecture. The considered feature extraction method reacts differently to the addition of noise to the test data. RAST-PLP shows robust performance and a great ability to handle noisy data compared to the other two considered feature extraction methods, namely PLP and MFCCs. The proposed approach (ESNEKMs) has also proven far more resilient to the presence of noise compared to other classification techniques, namely ESN and HMMs. This has encouraged us to promote a novel pipeline architecture where RASTA-PLP is combined with our proposed approach (ESNEKMs) to achieve superior performance in both clean and noisy environments.

5.4 AUDITORY NERVE BASED FEATURE FOR ESN

The previous findings of our conducted experiments have made it clear that data representation (the feature extraction method) has a major impact on system performance. This and the fact that almost all the state-of-the-art feature extraction methods have been developed to suit a particular classification approach, namely HMMs, has encouraged us to investigate the use of a new feature extraction method. The aim in developing more effective feature extraction techniques is being currently pursued from different directions, such as data compression oriented. Principle components analysis (PCA) is considered the main technique in this direction, and self-taught feature representation

uses unsupervised machine learning techniques to construct useful features from the raw data. Both of these techniques have their specific appeal and limitations. The main limitation of these two approaches is that neither allows developers to utilise the available human knowledge to construct the data representation. We argue that a biologically inspired direction is far more promising as it allows us to make use of the accumulating knowledge about the human auditory system. In addition, the reservoir computing approach is widely considered as a biologically-inspired classification technique which promotes such techniques in the preprocessing stage also to achieve a complete biologically-inspired pipeline architecture.

The use of biological systems to design models that perform cognitive tasks, such as speech recognition, is not a recently introduced idea. In fact, the approach that is used as the main block to compute the feature we adopt here was developed by Smith in 2004[87]. Despite the recognition by the research community of the potential of following a more biologically realistic approach in designing novel speech preprocessing methods, the lack of a classification approach that can handle such data representation prevents its adaptation in real-world tasks. This can be clearly seen on HMMs where the data representation needs to be non-correlated and highly compact (the input dimensional size needs to be typically between 13 to 39 to allows HMMs-based systems to achieve the desired performance). However, this very small size data representation is not critical in the reservoir-based approach which makes combining the two techniques highly appealing.

In this section, we built the first speech recognition system that adopts the Auditory Nerve Based Feature in the preprocessing stage and ESN in the classification stage. We compare the performance of the developed system with previously designed systems that use three different feature extraction methods and ESN. This evaluation is conducted in clean and noisy environments to investigate the effect of noise on the developed system. The following section

is organised as follows. We start by describing the conducted experiments and stating the aim of this work in the experiments section. The implementation details and optimisations phase is reported in the hyper-optimisaion section. In the results section, we present and compare the results of the conducted experiments and, in the following section (the discussion section), we analyse the performance of the model and consider the strengths and weaknesses of this architecture. Finally, we draw our conclusions in the conclusion section.

5.4.1 *Experiments*

The aim of these initial experiments is to investigate the potential for adopting the recently developed biologically-inspired acoustic signal preprocessing technique that is based on mimicking the auditory nerve in designing a reservoir-based Arabic speech recognition system. In order to achieve this aim, we developed a system that combines the two approaches in a single pipeline architecture and is evaluated on the Arabic Speech Corpus for Isolated Words. The selection of this corpus is based upon the fact that this is the only publicly-accessible Arabic corpus that is available in its raw format (see Chapter four for more details about this corpus). To evaluate the performance of the developed system, we compare it with other architectures that use three different feature extraction techniques, namely MFCC, RASTA-PLP and PLP. This evaluation is conducted in both clean and noisy environments, where different types of noise, namely white noise and babble noise, are added to the test set. These two noise types have been added to the test set in three different levels (30, 20 and 10 dB SNR). We also investigate the impact of the level of the auditory nerve feature on system performance to gain a better understanding of the dynamics of this new acoustic signal representation technique.

5.4.2 *Hyperparameters Optimisation & Implementation*

The developments models has been optimised following the same approach we have followed throughout this PhD study. The training sets have been divided into two subsets (training and validation). These two subsets are used to optimise all of the models' hyperparameters on a small size reservoir. Once the optimal values are found, we increase the reservoir size and optimise the regularisation parameter of the readout function to avoid any over-fitting that may result from using a large reservoir. The optimisation phase ends when the optimal value of the regularisation parameter is obtained and the system is then tested on the test sets. The test is repeated 10 times and the mean and standard deviation are reported for each model.

It is important to state that exactly the same sets and reservoir sizes are used in all of the developed and compared models to ensure a valid comparison between these different systems. Matlab code has been written to implement the ESN models, following the detailed description stated in [69], and the library developed by [27] is used in the preprocessing stage to compute MFCCs, PLP and RASTA-PLP features. In addition, a toolbox that has been developed recently by Smith is also used in the preprocessing stage to calculate the auditory nerve based feature (level 0 and the Gabor based feature from level 1 are used in the main model). Further comparison between the three different AN based feature levels has been conducted on a smaller reservoir size and only on the clean set due to several constants such as time and hardware limitations.

5.4.3 *Results*

In this section, we present the results of all of the conducted experiments and conduct a comparison between the developed systems. The results of all of the developed models are summarised in table 5.7, which shows a promising result for the model that adopts the auditory nerve (AN)-based feature in the preprocessing stage, when only AN and a Gabor filter are used. The AN-based model outperforms all of the state-of-the-art preprocessing techniques in the clean environments (no noise is added to the test set). The proposed model maintains this robust performance even when 30 dB NSR noise is added, regardless of the noise type. However, the suggested model performance differs based on the noise type when the added noise level is increased to 20 dB NSR and 10 dB NSR, as the model continues to outperform all of the other feature extraction methods with regard to babble noise and its performance is exceeded by MFCCs and RASTA-PLP on white noise. It is important to emphasise that the model outperforms the PLP-based model under all of the different considered acoustic environments (clean, babble noise, white noise and the different SNR levels).

Dataset		Feature Extraction	ESN
		MFCCs	98.97% (0.15)
	Clean	PLP	99.16%(0.11)
		RASTA-PLP	99.38 %(0.11)
		AN-Gabor	**99.78**%(0.08)
		MFCCs	98.03%(0.21)
	30 db	PLP	90.13%(0.36)
		RASTA-PLP	99.04%(0.11)
		AN-Gabor	**99.17**%(0.17)
White Noise			
		MFCCs	94.914 %(0.37)
	20 db	PLP	56.07 %(6.66)
		RASTA-PLP	**97.32** % (0.33)
		AN-Gabor	93.39%(0.55)
		MFCCs	77.19 %(2.12)
	10 db	PLP	19.83 %(3.83)
		RASTA-PLP	**87.48** %(1.47)
		AN-Gabor	34.07%(3.85)
		MFCCs	97.23 %(0.29)
	30 db	PLP	97.87 %(0.36)
		RASTA-PLP	99.22 %(0.19)
		AN-Gabor	**99.46**%(0.14)
Babble Noise			
		MFCCs	89.72 %(0.87)
	20 db	PLP	89.47 % (2.43)
		RASTA-PLP	97.18 % (0.42)
		AN-Gabor	**97.27** %(0.41)
		MFCCs	64.12 % (2.31)
	10 db	PLP	56.23 % (4.82)
		RASTA-PLP	**85.45** % (8.6)
		AN-Gabor	66.56 %(1.15)

Table 5.7: The results obtained by auditory nerve based and the other compared systems, we report the mean over 10 runs and the standard deviation. The compared results from Table 5.4.

The results shown in table 5.7 encouraged us to investigate different combinations of the AN-based feature levels which are used as the input of the ESN classifier. Seven models have been constructed to cover all of the possible combinations of the three AN-based features, namely AN, Gabor and Onset. The results of this comparison are summarised in table 5.8. The best performance is obtained by the model that combines a level 0 feature (AN) with ESN. The use of the higher level feature (Gabor and Onset) decreases the accuracy of the system and increases its complexity. The use of the Gabor filter-based feature degrades performance slightly, by about 1%. The best accuracy rate is 99.27%, which is reported on the AN model, where the onset model achieved the lowest performance, with a 66.23% accuracy rate. This difference in performance emphasises the impact of the selected feature extraction technique on system performance.

Systems	Accuracy Rate
AN	**99.27 % (0.16)**
AN and Gabor filter	98.78% (0.17)
Onset and AN	98.36% (0.12)
Gabor filter	97.74% (0.37)
Onset,Gabor filter and AN	97.15 %(0.30)
Onset	66.23% (0.59)
Onset and Gabor	96.05 %(0.23)

Table 5.8: The results of investigating the performance of the different models that constructed by all the possible combination of the AN based feature levels.

5.4.4 *Discussion*

The AN-based approach has proven to be a very promising technique when combined with ESN. This can be seen from its superior performance on clean sets and its competitive results on noisy sets. It is also clear that. adding a high level of noise to the signal degrades performance considerably, which is the main limitation of this technique. The hierarchical feature paradigm does not improve system performance but in fact decreases the accuracy rate while increasing the complexity of the system (this includes computing these higher level features and the added complexity due to the increase in the reservoir input dimension). However, the main idea of using a more realistic, biologically-inspired preprocessing technique in a speech recognition system is very promising and we encourage the research community to investigate this paradigm in more depth based on our findings in this work. Finally, it is important to emphasise that the other compared state-of-the-art feature extraction techniques have been developed over several decades by many scholars who devoted their efforts to designing and improving these technique whereas the AN-based feature outperforms all of these compared techniques in clean and low level noise. This clearly demonstrates the proposed pre-processing potential.

5.4.5 *Conclusions*

We have investigated the potential of developing an Arabic speech recognition system based on ESN and recently introduced a more realistic, biologically-inspired preprocessing technique, namely the Auditory Nerve (AN)-based feature. The developed system has been evaluated on the Arabic Speech Corpus for Isolated Words. The model performance under different types

and levels of noise has also been investigated. In addition, a comparison has been conducted between the developed system and other baselines systems that use a state-of-the-art speech recognition system namely MFCCs, PLP and RASTA-PLP. The results show the superior performance of the developed model in noise-free environments and in the light presence of noise. The proposed model has also shown competitive results on high level of noise environments. These results have encouraged us to study the hierarchical AN-based feature levels in more detail so 7 models have been implemented to cover every possible combination of the three different feature levels. The results suggested that using an AN-based feature without the higher level features improves performance. These results are very encouraging and we feel very excited about developing this system further, beyond the constrains of this PhD study.

6

DISCUSSION AND FUTURE WORK

6.1 INTRODUCTION

Throughout this work, we have investigated the potential for adopting reservoir computing for an Arabic speech recognition system. Several experiments have been conducted on several corpora, including a self-developed corpus which is currently the largest corpus of Arabic isolated words, to evaluate the use of the conventional ESN approach. The developed corpus has allowed us to investigate the system performance under different feature extraction techniques (see Chapter four). The superior performance reported by these experiments has encouraged us to improve the ESN approach which has resulted in the development of two ESN-based novel approaches, namely ESNSVMs and ESNEKMs (see Chapter five). The effect of noise on the developed approached performance is also studied in Chapter five. The observed impact of the selected feature extraction method has promoted us to investigate the potential for adopting a recently-developed, biologically-inspired preprocessing approach. In this chapter, we discuss all of the findings of the experiments conducted throughout this PhD study. The implications of these findings and the suggested future work directions are also covered. In addition, the strengths and limitations of this work are discussed.

Each section in this chapter discusses in detail a single component of the ESN-based speech recognition system. The finding related to the considered feature extraction methods are discussed in the first section as this is the first component of the system. Secondly, the components of the classification

approach are covered which include the input layer, reservoir layer, activation function and output layer. The effect of noise on system performance is also discussed in a separate section and finally we conclude this chapter with a summary.

6.2 FEATURE EXTRACTION METHODS

The role of the selected feature extraction method on system performance has been investigated for all the developed models. Three main state-of-the-art preprocessing techniques have been considered in this work; namely, MFCCs, PLP and RASTA-PLP. These methods have been adopted and compared across the conventional ESN approach, two introduced novel systems (ESNSVMs and ESNEKMs) and HMMs baseline models. These experiments have been conducted on three different corpora and, in the presence of a stochastic element, the experiment is repeated at least 10 times to ensure reproducible results and valid conclusion. In addition, a recently- developed, biologically-inspired preprocessing technique, namely AN, is adopted to develop a conventional ESN based Arabic speech recognition system. It is clear from the results of all of the conducted experiments that the selected feature extraction technique has a major impact on system performance. This has been observed under the use of all of the considered classification approaches, namely ESN, ESNSVMs, ESNEKMs and HMMs, across the two different corpora. It important to emphasise that the SAD corpus is not used to investigate the effect of the selected feature extraction on system performance as it is only available in preprocessed format (MFCCs). Our findings in Chapters four and five suggest that the impact of the selected feature extraction method differs according to which method is adopted. This promotes the design and testing of the complete pipeline architecture to capture this interaction between the preprocessed approach and the classification technique.

6.2.1 *State-of-the-Art Feature Extraction Methods (MFCCs, PLP, RASTA-PLP)*

In this section, we focus on our findings regarding the use of these three feature extraction techniques across all of the different classification approaches and the various corpora. We discuss these findings in depth and emphasise the strengths and limitations of these methods and possible future research directions. In our investigation of the role of feature extraction in system performance, an intensive empirical approach has been followed and several models have been implemented to evaluate different pipeline architectures. Our main finding is that the feature extraction method has a significant impact on system performance, which is not maintained in all of the various regimes. In other words, the selection of the feature extraction method is crucial to achieve the best possible performance but these feature extraction methods provide different performances under different settings. A good example is RASTA-PLP, that provides superior performance compared to the other techniques, given that the acoustic signal is long enough to implement the RASTA filter. This has been demonstrated by our experiments on our developed Arabic speech corpus for isolated words as, where the audio file is long enough to implement, the RASTA filter and the RASTA-PLP outperform the other approaches but when the same models are evaluated using the Arabic phonemes corpus, where the acoustic signal is far shorter, the performance of RASTA-PLP has failed dramatically whereas the PLP approach has achieved the best performance. In addition, the performance of a specific feature extraction technique combined with a classification approach should not be used as a hard estimate of the performance of the same feature extraction method combined with another classification method. This means that, in order to design a robust system, different pipeline architectures need to be tested on the available data to ensure the selection of the most suitable one for the regime at hand. We also find major differences in terms of performance

between the considered feature extraction approaches with regard to their robustness to noise. These differences are discussed in depth in section 6.7.

There are two main possible future directions that can be pursued and investigated. The most obvious direction is to study the effect of combining two or all three different feature extraction methods under different classification approaches. Although this would be a computationally expensive approach, the potential increase in performance could justify its use. To the best of our knowledge, this has been investigated neither with regard to an ESN-based speech recognition system in general nor an Arabic speech recognition system in particular. This approach is supported by the fact that ESN can handle large input dimensional space without losing its robust performance, unlike HMMs.

Calculating the dynamic features for each technique, the first and second derivative is widely adopted in other state-of-the-art techniques which would suggest that such adoption could improve system performance. In theory, recurrent connections allow the ESN-based model to capture the dynamic of the system but recent results in[92] suggest that adding the first and second MFCCs derivatives does improve system performance. Testing this claim on the MFCCs and other techniques looks promising to increase the system performance even further.

6.2.2 *Auditory Nerve Based Feature*

The findings of the Auditory Nerve (AN) based feature appear promising. This recently-developed technique outperforms all of the state-of-the-art techniques considered in this study. However, this robust performance decreases significantly when noise is added to the test set (further discussion on the effect of noise is presented in section 6.7). The comparison between the three AN- based features, namely AN, Gabor and Onset, showed that AN is by far

the most effective. The addition of Gabor features reduces the performance slightly whereas the use of Onset causes a major decline in the accuracy rate.

The major finding of this work is that an AN-based feature and particularly level 0 feature (AN) has great potential and can provide superior performance in a noise-free environment compared to the well-known state-of-the-art techniques (MFCCs, PLP and RAST-PLP). This means that the use of a biologically-inspired preprocessing technique can increase the overall performance of an Arabic speech recognition system. Higher level features, Gabor and Onset, are not useful and can have a negative impact on system performance while also increasing the model complexity, although this has been tested only on clean sets.

There are many future research directions that can be pursued and we will state the most promising areas. Optimising the current implementation of the system to increase its speed is an important direction that can lead to wider adoption in the field specially for real-time applications. The weak performance of the developed system needs to be addressed to open the door to using this technique in real-world applications, where different types of noise and acoustic environments are present. This can be achieved by adopting the RASTA filter or other established technique or going back to the biological system to see how this is being accomplished. In addition, a comparison using noisy data between AN, Gabor and Onset is needed to investigate the effect of noise on each level. Finally, a hybrid system that combines AN and other state-of-the-art techniques has the potential to combine the strengths of the two worlds, as the redundant information is very limited due to the intrinsic differences when computing the features between these techniques.

The input layer is the first component of an ESN system that maps the input data, the extracted feature, to the reservoir dimensionality. This mapping is conducted using a randomly-generated matrix and a scaling parameter. This parameter is typically optimised using a validation set to find the value that offers the best performance. In this work, we have investigated replacing this random process with a more systematic procedure using a convolutional approach. Several Gabor filers have been used to convolve the extracted feature and map it to the reservoir dimension. The main issue in this approach is how to determine the value of these Gabor filters, or any other filter. The use of randomly initialised Gabor filters drives us back to the stochastic nature of this layer and provides poor performance while increasing the complexity of the system. On the other hand, attempting to learn these parameters using the data, by back-propagation, requires far more computational power, which makes it impractical in many situations. Our main findings regarding this layer is that using the conventional approach to generate and optimise this layer, randomly constructed and scaled to obtain the best performance, is the most efficient method. In addition, we have found that optimising the scaling of this layer is highly related to the leakage rate and the scaler of the reservoir layer. This means that using a greedy approach, one parameter at time, to optimise these three parameters results in poor performance. Thus, we highly recommend optimising these parameters jointly using a grid search or other optimisation technique e.g. a genetic algorithm.

Despite our limited success in improving the construction process in this layer, we still believe there are many possible architectures that can improve the overall performance. The main paradigm and the most promising is a random process by cluster mapping where the extracted feature is mapped to the required dimension using a clustering technique. Soft clustering algorithms

such as a multivariate gaussian mixture (which can be constructed using expected maximisation) or hard cluster techniques such as K-means can be used to map the extracted feature to the reservoir size using a number of groups that equals the reservoir size. This allows the input layer to play a far more active role in capturing the dynamic of the data and make use of any unlabelled data, as training the cluster is conducted in unsupervised mode.

6.4 RESERVOIR LAYER

The reservoir layer is the most crucial component of the system, as it controls the system's memory capacity. This memory is provided by the recurrent connection between the nodes which allows the model to capture the dynamic behaviour of the data. This layer has two main hyperparameters that need to be optimised in order to obtain the desired performance. The first and by far the most important hyperparameter, that has the highest impact on system performance, is the leakage rate, which controls the fading memory capacity of the system; it takes values in a range 0 and 1. Based on our work, we recommend devoting a substantial amount of time and resources to optimising leakage rate, as selecting a poor value prevents the reservoir from modelling the time dependencies of the data.

The other hyperparameter that needs to be optimised in this layer is the scaling factor that scales the weights of the recurrent connections. These recurrent weights are initialised randomly and then the scaling factor is applied to drive the network to the desired behaviour. This parameter is typically optimised in grid search mode in conjunction with the input scaler and the leakage rate. It is important to state that, under the conventional approach, it has been suggested that a sparse random connection is required, which means that another hyperparameter, that controls the sparsity of the weights matrix, needs to be optimised. The use of a sparse connections scheme

is supported by observations from neuroscience which suggest that the neural connections in the human brain are very sparse. However, this work and other recently-published studies have found that the sparse connection does not influence system performance and that using a dense matrix has the advantage of reducing the model complexity by avoiding the need to optimise the sparsity hyperparameter.

Future work may include investigating different types of recurrent connections patterns and their effects on system performance. The possibility of using parallel reservoirs with different leakage rates is another exciting direction to pursue. The use of multiple reservoir architecture has the potential to provide a rich memory capacity with a far lower model complexity, and the overall parameters can be significantly smaller. This will reduce the need to optimise and run the reservoir-based system while also improving performance.

In summary, our main findings regarding this layer fall on the optimisation procedures where we have developed a systemic approach that reduces the CPU time, memory usage and system complexity needed to optimise the reservoir. This approach focuses on finding the optimal value for the leakage rate with a small size reservoir using a grid search that includes the input scaler and the recurrent weights scaler. Once the leakage rate has been optimised on this small reservoir, a larger reservoir can be used to improve the system accuracy. Two main potential future research directions have been discussed, which include new recurrent connections patterns and parallel reservoir architecture.

6.5 ACTIVATION FUNCTION

In this work, we have studied the role of the selected activation function on the performance of the conventional, ESN-based Arabic speech recognition system and our developed approach, the ESNEKMs-based Arabic speech recognition

system. The two most widely used activation functions are considered in this work, namely tanh and sigmoid, under two different feature extraction methods, MFCCs and PLP. This meant that eight models have been implemented to investigate all of the possible architectures. These experiments have been conducted on the Arabic phoneme corpus. Our main findings suggest that there is a significant impact on system performance. In addition, we have found that there is a relationship between two of the considered factors, namely the activation function and the feature extraction method. In other words, to obtain the best performance, these two selections need to be made in conjunction, as the performance of the system under a specific activation function differs when another feature extraction technique is applied.

The activation function is adopted to enable the network to model complex behaviour. In the early days of the field, when the back-propagation algorithm was introduced to train the Multilayer perceptrons (feedforward) network and its modified version, back-propagation through time, was also proposed to train a network with recurrent connections, that provides the system with a form of memory to handle dynamic behaviour, very limited types of activation function were used. This was mainly due to the fact that the error back-propagation learning paradigm requires computing the derivative of the applied activation function. This meant that the derivative of the activation function needs to be easily computed to ensure feasible, efficient learning. For this reason, only two types of activation function have been widely adopted, namely sigmoid and tanh, as their derivative can be easily computed. We believe that it is important to emphasise that, in the conventional artificial neural network and even in the conventional reservoir computing approach, no specific activation function was proposed. In the two approaches, the only property needed in the activation function is strictly increasing non-linearity. These two types of activation function have been adopted in the reservoir computing research community without solid justification, as the condition of

an easily computed derivative is not applied in this approach. This is due to the fact that the reservoir computing learning paradigm does not back-propagate the error through the network which means that any non-linear activation function can be used, in theory. Thus, we believe that using new activation functions is a major area for future work, particularly when considering the significant impact of the activation function found in this work. Another potential area is investigating the robustness of these activation functions in noisy environments. The output of such studies will enable scholars to design far more noise resilient models which is a crucial property for real-world application.

6.6 OUTPUT LAYER

The output layer is the only learnt layer in the reservoir computing structure. The learning process is conducted under the conventional approach by a linear classifier that takes the reservoir response as input and maps it to the target labels. The use of a linear classifier in the output layer meant that a larger reservoir is needed in order to find a linear decision boundary that separates the different classes. However, using a large reservoir is very computationally expensive, as the complexity of the system grows as the square of the reservoir size, which can lead to hardware limitations issues. In addition, depending on the data available to train the read-out functions, using a large reservoir can also lead to over-fitting of the training data, which degrades the generalisability of the model. This and the fact that the learning process is in general challenging, regardless of the algorithm used, has encouraged us to focus on improving the conventional approach by developing the output layer further and investigating several potential architectures that address the discussed issues. This effort resulted in the development of two novel approaches, namely ESNSVMs and ESNEKMs. These two techniques were introduced to

the research community in [2] and [3] . In the following sections, we discuss each approach individually and conclude by suggesting new directions for future work.

6.6.1 *ESNSVMs*

This technique is the result of our first attempt to improve the output layer in the conventional reservoir computing approach. It proposes replacing the simple linear read-out with a support vector machines (SVMs) classifier which significantly increases the classification capability of this layer. The selection of SVMs rather than other non-linear learners such as Multilayer perceptrons is based on its attractive properties that allow us to maintain most of the reservoir computing advantages. These include a convex optimisation solution that ensures a high level of reproducibility and fast learning process. In addition, it avoids any learning problems that result in obtaining a non-invertible matrix as the reservoir response which prevents the use of a typical pseudo-inverse solution to compute the linear decision boundary. Finally, a new kernel can be developed to tackle the task at hand which ensures a higher level of flexibility and classification power. The SAD corpus has been adopted to implement and evaluate this novel approach and the results are compared with the conventional ESN model. In these experiments, we have also studied the effect of reservoir size on system performance.

Our findings suggest that this novel approach outperforms the conventional ESN technique. The margin between the performance of these systems increases as the size of the reservoir reduces. This might be due to the fact that the higher data dimensions lead to a higher probability of finding a linear decision boundary. Thus, the linear read-out function can improve its performance on a large reservoir size (higher dimension) and reduce the margin between the two approaches. Thus, although both of the techniques' perform-

ance decreases with a reduction in reservoir size, ESN is far more sensitive to such changes. This means that a large reservoir is needed to obtain state-of-the-art competitive results with ESN; however, in ESNSVMs, the desired performance can be obtained with a far smaller reservoir size.

However, there are some limitations that prevent the degradation of the demonstrated superior performance under specific regimes. The first and major limitation is related to the binary nature of SVMs, which means that a large number of SVMs classifiers are required to tackle the multiclass classification task. This increases the complexity of the model and makes its use impractical for regimes where a large number of classes need to be separated. This issue is not only related to the training phase but even to the test phase, as each sample needs to be classified by all of the constructed SVMs binary classifiers and then a majority vote process is conducted to classify the sample. Another limitation is related only to the training phase, as adopting SVMs in the output layer means that there are more hyper parameters, Kernel types and other SVMs parameters that need to be optimised, which increases the complexity of the learning process.

Future work includes applying this approach for new application domains such as video and non-speech acoustic signal classifications. The regimes in these suggested applications are ideal for the ESNSVMs' characteristics which results in efficient and superior performance. In general, ESNSVMs has proved to be very effective in a regime where the input dimension is very large which makes it very difficult to map it to even a higher dimensional space using the reservoir approach. This is not only problematic from the computational point of view but also from the number of samples that are needed to train a classifier with such a large degree of freedom. In addition, the number of classes in this type of application tends to be small which suits the binary nature of SVMs. The effect of different types and levels of noise also needs to be investigated to obtain a better understanding of the strengths

and limitations of the proposed approach. Finally, other future work areas include the design of new kernel types that aim to improve the performance of reservoir computing-based systems.

6.6.2 *ESNEKMs*

The development of the ESNEKMs approach is based on their superior performance. The ESNSVMs technique shows the potential for increasing the read-out layer classification capability; however, its main limitation in handling tasks with a large number of classes imposes a barrier against its wider adoption within the community. Thus, we have developed and introduced the ESNEKMs approach to address this main weakness in ESNSVMs while maintaining the robust performance. In this novel approach, we combine two recently developed approaches to train neural networks, that both avoid the use of an error back-propagation learning paradigm. Only the output layer is trained in these two techniques which means that the time required to train the system is reduced. The proposed approach combines the strengths of both techniques as it provides memory for the EKMs that enables it to handle dynamic behaviour and provide ESN with a non-linear classifier that has the following attractive properties: fast to train, a convex cost function and a much more robust performance compared to the linear read-out function suggested under the conventional ESN approach. This novel approach has been evaluated with three different corpora using three different feature extraction methods. The effect of reservoir size on system performance is also considered in this evaluation. The obtained results have been compared with those of several published studies, and the ESN, ESNSVMs and HMMs models. A comparison has also been conducted in the presence of different types and level of noise to study the effect of noise on performance and investigate the potential of adopting this approach in real-world applications.

Our findings suggest that ESNEKMs offered a significant increase in the performance with a small reservoir size compared to ESN and ESNSVMs. We have shown that an ESNEKMs model with 100 nodes outperforms ESN and ESNSVMs, both of which used a 10 times larger reservoir, a 1000 node reservoir. This clearly demonstrated the superior performance of the developed technique. However, when a very large reservoir size is used, the margin in performance between the three techniques decreases. This may be due to the fact that the linear read-out function and the SVMs classifiers have a limited classification ability and they need a far higher mapping in order to provide state-of-the-art performance compared to EKMs. However, even when a larger reservoir is used and the performance margin decreases, the ESNEKMs approach provides a more stable performance that can be the results of the robust EKMs classifier that are used in the output layer, that allow the model to reduce the variation in the performance due to the random initialisation of the reservoir.

The main limitation of this approach is the added complexity in the output layer. Optimising the EKMs classifier, kernel type and its parameters, increases the time required to train the system. Another major limitation is the selection of the kernel size as, under the conventional EKMs approach, all of the training samples are used to construct the kernel matrix which means that the size of the kernel matrix increases as the square of the sample size. A good example is that, suppose we have a corpus containing 1,000,000 samples, the dimensionally of the resulting kernel matrix is 1,000,000 by 1,000,000. This is very expensive computationally not only in the training phase but even in the test phases, where each single test sample needs to be mapped by this huge kernel matrix.

Future work includes applying this novel approach in new application domains such as video and non-speech acoustic signal classification. In addition, there is a need to develop a new approach to enable this technique to be

scaled up to incorporate a huge number of samples. This could be done by investigating different paradigms to select a subset of the available samples to construct the kernel matrix. These paradigms include random selection and adopting clustering algorithms to find the best subset that represents the data and the online learning paradigm. Finally, another promising area is investigating new pipelines architectures that combine ESNEKMs with novel extraction techniques.

6.7 THE EFFECT OF NOISE

Noise resilience is a crucial property in developing a speech recognition system for real-world applications. In a noise-controlled environment, the task is far easier compared to real-world scenarios where the acoustic environment is not controlled. Hence, it is important to evaluate the novel system under different types and levels of noise to estimate its performance in the real-world. Thus, this work has devoted significant efforts to testing and evaluating the developed systems under different noise types and levels. This investigation includes four types of feature extraction methods; namely MFCCs, PLP, RASTA-PLP and AN bases features. It also includes three different classification approaches: ESN, ESNEKMs and HMMS. In this section, we discuss our findings and suggest new directions for future work.

Our findings suggest that feature extraction and the classification approach have both had a major impact on the model performance in noisy environments. The sensitivity to the addition of noise in the test set varies remarkably from one pipeline structure to another. In addition, the model performance in clean environments must not be taken as an indication of its behaviour under noise. A good example is the developed models that adopt the PLP technique, which offers a very competitive accuracy rate, roughly 99.31% when combined with ESNEKMs, on the clean set but drops dramatically to 35.35% when a high

level of white noise is added. Thus, we emphasise the importance of following a comprehensive approach when designing noise robust speech recognition system that consider the complete pipeline architecture. Our findings also suggest that adopting reservoir methods in developing speech recognition systems can improve performance. In particular, adopting ESNEKMs has proven highly effective and helped the system to strongly improve its performance in noisy environments. Finally, we promote a wider adoption of the architecture that combines RASTA-PLP and ESNEKMs as it has the best performance under all noise types and levels which makes it very attractive for real-world application.

The AN based feature has shown superior performance on the clean set and promising results on noisy data. It outperforms all of the other techniques when no noise is added and continues to provide higher performance compares to the PLP approaches under all of the different types and levels of noise. However, RASTA-PLP provides far better performance in the presence of a high level of noise, particularly 10 dB white noise.

There are many promising areas that can be investigated to develop the work present in this PhD study. The main area is to work on improving the AN-based feature either by adopting a RASTA filter or developing novel techniques that allow the AN based feature to maintain its superior performance in noisy environments. Another area is investigating the effect of combining different types of feature extraction method on the system's resilience to noise. Finally, investigating a novel pipeline architecture that takes advantage of the ESNEKMs approach has serious potential. This includes developing a novel architecture that combines ESNEKMs and the AN-based feature and evaluating its performance in noisy environments.

Every study has some limitations that are imposed by different constraints, such as time and financial constraints. This PhD study is no exception, as the objectives of this work need to be achieved within a firm deadline and the very limited resource devoted to this project. Beyond these typical constraints, there are some main limitations that are particularly imposed on this work. The main limitation of this work arises from the fact that the Arabic language is the official language of 26 countries and so has a variety of accents. In some cases, the variation is so high that it is classified as a dialect, so covering all of these variations is well beyond our limited resources. In our future work, we would like to establish collaboration across the Arab world to create the largest Arabic corpus for large vocabulary continuous speech to date, covering all of the major Arabic accents and dialects, and release it freely to the community. The importance of developing such a valuable resource cannot be exaggerated, as its introduction to the research community would help to advance this field far faster and remove a major barrier that prevents scholars from participating in the field. In addition, it will help to reduce the use of in-house corpora in developing and evaluating new systems which will increase the quality of the conducted studies by allowing scholars to focus on the main aspects of the work instead of on corpus development. In addition, using such a corpus means ensuring a high level of reproducibility which will also increase the quality of the conducted work

The backbone of modern speech recognition systems are the available data, as all the state-of-the-art techniques construct a model and learn from the training samples. The crucial impact of an available corpora to train the model on system performance is not only limited to the speech recognition application domain but also covers all machine learning applications. In this work, we have developed the largest Arabic isolated word corpus that contains about

10,000 utterances spoken by 50 speakers, which in itself was a serious challenge. The process of developing such a large corpus took months of hard work, which eventually increased the duration of this study but it was a necessary step to accomplish the objective of this study. We have released this corpus to the research community and plan to increase its size and improve certain aspects of it, such as providing more balanced, representative participants with different accents, ages and genders.

The other main limitation of this study is that a continuous speech system is not included. This is due to the fact that there is no phoneme labeled Arabic continuous speech corpus. Arabic continuous speech recognition systems have great potential in the commercial world as well as within academia. In such a system, there is a need to develop and integrate a language model to improve the system and achieve the desired performance. We believe that a reservoir-based system can be used to develop the required language mode with great success, based on its superior performance in constructing the acoustic model demonstrated in this study. However, significant efforts are required to develop such a system and confirm this claim, which we will also attempt to make in the near future.

The main challenges are related to the fact that reservoir-based systems were developed only recently, which means there are very limited resources available to scholars. This and the fact that the Arabic language is also considered as a poor resource language meant that we needed to develop our own data. The other main challenges stem from the theoretical aspect of this study, as designing two novels approaches, ESNSVMs and ESNEKMs, resulted in an extensive investigation that aimed to explore the characteristics of and assumptions made by the different state-of-the-art classification techniques. This investigation allows us to develop and justify our novel approaches and ensure that we have a solid understanding of the behaviour of the different components. This is crucial in discussing and evaluating the proposed ap-

proaches in adequate depth and identifying the weaknesses, strengths and the ideal regimes where adopting a certain approach is expected to be highly successful for each model.

6.9 CONCLUSIONS

In this chapter, we discussed the findings of our PhD study and stated its limitations and strengths. A variety of future work directions have been also suggested and its potential impact has been considered. To ensure an in-depth coverage to all of these aspects, we considered each component of the ESN approach separately and focused on improving the complete approach by developing every specific component. We also revisited our research objective and discussed the design of each experiment individually and how it is related to the accomplishment of these objectives. Pointers to previous work in the literature have been stated where needed to provide a broader context for our work and we have critically reported the status of the Arabic speech recognition community and our suggestions about how it might be improved.

Our two novel approaches, ESNSVMs and ESNEKMs, were also discussed in detail and we emphasised the different regimes under which each model is expected to provide superior performance. The impact of the selected feature extraction method is also stated and its implication in designing a new robust Arabic speech recognition system is highlighted. The performance of ESN, ESNSVMs, and HMMs under different feature extraction methods (namely MFCCs, PLP, RASTA-PLP and AN-based feature), in the presence of a variety of noise levels and types, is analysed and the importance for noise resilience speech recognition systems is justified. In this next chapter, we focus on the implications of this work conducted for this PhD study combined with an overall summary of it.

7

CONCLUSIONS

In this chapter, we provide a brief summary of this work and revisit the research objectives to discuss how they have been accomplished by the findings of this study. In addition, we discuss the implications of this work and finally draw our conclusions.

7.1 SUMMARY

This PhD study has investigated the potential for adopting the recently introduced reservoir computing technique to develop an Arabic speech recognition system. We began this thesis by specifying the research objectives and emphasising the potential impact of this work and stating the original contributions of this study. Automated speech recognition systems are introduced in the second chapter to provide the necessary context for the presented work. This includes a basic definition of automated speech recognition systems, a brief introduction to human speech and descriptions of the four considered feature extraction methods, namely MFCCs, PLP, RASTA-PLP and AN based features. The feature extraction method is the first component of any state of the art speech recognition system. This enabled us to shed light on the differences between the various methods and emphasise the importance of this phase, as using a weak feature extraction method would lead to far poorer performance. In addition, we have emphasised that many of these models borrow concepts from the biological system whereas AN based systems seek to mimic it in relatively more detail.

Chapter three was devoted to the other component of modern speech recognition systems, namely classification. The machine learning field is introduced in this chapter together with many crucial concepts related to the field, such as the need for generalisation, over-fitting, no-free lunch theorem and Occam's razor. There are a variety of machine learning approaches, making it impractical to attempt to cover all of them in this work and also we believe many of these methods are not directly related to this study. Thus, we restrict our discussions to the classification methods that are used and include variations on the considered approaches when necessary. The main theme of this thesis, reservoir computing, is introduced and its history, concepts, main techniques and attractions are stated. We also emphasise the differences between the static and dynamic classification approaches to demonstrate the potential of adopting ESN.

The first reservoir based Arabic speech recognition system is introduced in chapter four. This system adopted the ESN approach, which has been described in chapter three, and a variety of feature extractions methods, introduced in chapter two. This chapter also covers descriptions of all of the considered corpora. This includes our self-developed corpus that contains about 10,000 samples, making it the largest Arabic speech corpus for isolated words. It has 50 participants. The motivation behind the development of this corpus is also stated along with a list of all of the uttered words. Each section of this chapter is devoted to reporting the experiments conducted on a single corpus. This allows us to create a coherent argument and justify our evaluation method. For example, in the SAD corpus, we evaluate our system in light of the other published models and only the MFCCs feature extraction method is used. This is because SAD is a public corpus that several studies have used to evaluate their proposed approaches, and because SAD is only available in preprocessed form, MFCCs vectors. However, in the Arabic speech corpus, for isolated words, we evaluate the system by developing a

state-of-the-art baseline model using HMMs, as this is a new corpus and no previous work has been conducted on it. In addition, applying different extraction methods is made possible because the corpus is available in its raw format instead of a preprocessed format. The developed approaches that adopted ESN outperform all of the other compared systems over all three corpora. This means that the ESN-based model provides superior performance compared with the considered published model found in the literature and the HMMs baseline models. These results demonstrate the potential of adopting ESN for an Arabic speech recognition system, which has encouraged us to work to improve the system further.

In Chapter five, we built upon the success of the reservoir-based Arabic speech recognition systems demonstrated in Chapter four. Two novel approaches, ESNSVMs and ESNEKMs, are presented and evaluated. The first proposed approach, ESNSVMs, is implemented by improving the classification capability of the output layer of the conventional ESN approach. This has been accomplished by replacing the linear read-out function used in the ESN by an SVM classifier. The evaluation of this approach shows that it offers superior performance compared to ESN and state-of-the-art published models, that used the same corpus to evaluate the performance, thus making this a valid comparison. The improvement in the performance increases as the reservoir size decreases and state-of-the-art competitive performance can be achieved with a far smaller reservoir size. ESNSVMs also maintain all of the main attractions of the reservoir computing paradigms, including a convex optimisation solution to the cost function used in the output layer. Only the weights on the output layer are learnt which enables fast training and the ability to capture the dynamic behaviour found in the data. Other strengths of the developed approach include enabling the researcher to use or design a kernel that is suitable to the task at hand and avoiding the risk of obtaining a non-invertible matrix in the reservoir response, which is problematic when

the pseudo inverse method is applied in the output layer, which is the typical approach in ESN. However, all of these advantages come at the cost of increasing the complexity of the output layer which is reflected in the more overall complex model. This increase in complexity arises mainly from the need to optimise the kernel hyper parameters that are applied in the output layer such as kernel types and their related parameters and the cost coefficient that controls the model complexity of the constructed SVMs classifiers. Another major limitation of this model is the binary nature of SVMs which means that this model is only feasible in regimes with only a small to medium number of different classes. However, this limitation has been addressed in the other novel model (ESNEKMs).

The ESNEKMs approach, which is also covered in this chapter, has been developed to address the limitation of the ESNSVMs. The motivation behind the development of the model and the reasons that contributed to the selection of an extreme learning machine in the output layer are also stated. This proposed approach overcomes the main limitation of ESNSVMs, which is handling tasks with a larger number of different classes. This property is crucial for building a speech recognition system that can be deployed beyond a small number of isolated words in a word-based system and can cover all of the phonemes of a specific language, in a phoneme based system. Thus, ESNEKMs provides superior performance compared to ESN and ESNEKMs while maintaining the abilities needed to develop any type of speech recognition system. This is important to promote a wider adoption in the field and demonstrate the strengths of the reservoir-based system in general. An extensive evaluation of the proposed system, ESNEKMs, over the three different corpora has been conducted that includes comparing the system to the ESN, ESNSVMs, HMMs baseline models and other published models found in the literature. The results show the superior performance of ESNEKMs as it outperforms all of these compared models across the considered corpora. In addition, the suggested

model has proven capable of outperforming these compared models with a far smaller reservoir size, as ESNEKMs with 100 nodes outperforms the ESN and ESNSVMs models with 1000 nodes. This is achieved while maintaining the same attractive properties of ESNSVMs, such as allowing researchers to select or design the kernel that models the data most effectively while keeping the convex optimisation solution in the output layer which leads to a more reliable performance and reproducible results. A variety of state-of-the-art feature extraction methods are applied to investigate the effect of the selected representation method on system performance. This also allowed us to find the best pipeline system, the complete architecture that contains the feature extraction method and the classification approach.

The effect of noise on the performance of the system has also been investigated using the ESN, HMMs baseline models and the proposed approach. This has been conducted by introducing different types and levels of noise while the training phase is performed using clean data. The results show that RASTA-PLP is by far the most noise robust feature extraction method and the ESNEKMs is the most resilient to noise among the considered classification approaches. This means that the best pipeline architecture is the one that combines RASTA-PLP and ESNEKMs. In addition, the ESN-based approach shows robust performance compared to the HMMs which shows the potential for adopting the reservoir-based approach for real-world applications where a high level of noise and different acoustic environments tend to be present.

In addition, we introduce the AN based feature to the automated speech recognition field, the Arabic domain in particular. The use of a more realistic biological inspired model with a reservoir-based speech recognition system has proven highly successful. The pipeline architecture that contains the AN based feature and ESN offered superior performance compared to other architectures that use state-of-the-art feature extraction methods (RASTA-PLP,MFCCs and PLP). This robust performance was mainly due to the fact

that reservoir-based speech recognition systems can handle input data with a large number of dimensions, unlike other conventional methods, such as HMMs. However, there are two main limitations associated with applying this recently-developed feature extraction method. The first limitation is that the performance of the system degrades badly when a high level of noise is added, especially white noise. This creates a serious barrier that prevents the adoption of this technique in real world applications. The second limitation is that the time needed to compute these features is relatively longer than the standard methods, which makes applying this technique unfeasible for real time applications. These two main issues need to be addressed to ensure a wider acceptance in the field and a broader range of possible applications. Finally, these promising results have encouraged us to investigate the different levels of AN features to study the contributions offered by each level. The findings suggest that the level 0 feature is the most effective level and that adding the other levels has a negative impact on performance.

In Chapter six, we discussed the findings presented in Chapters four and five and proposed future work directions to build upon the outcomes of this work. We began this chapter by discussing the impact of the feature extraction methods on system performance. This includes the strengths and weaknesses of each feature extraction method. The significant influence of the selected feature extraction method on system performance has been emphasised. In addition, we have suggested that the performance of a selected feature extraction method on a specific pipeline architecture, combined with a particular classification method, should not be treated as an accurate medium for estimating the performance of the same feature extraction method combined with other classification approaches and a different pipeline architecture. The potential of the biological inspired feature extraction technique, namely the AN-based feature, is also discussed. The superior performance obtained in noise-free acoustic environments makes a wider adoption very appealing. In the feature

extraction domain, we have proposed several directions for future studies that include the following. Firstly, experimenting with new feature extraction methods, as using the traditional, standard approaches that have been dominating the field mainly due to their convenience is no longer justified with reservoir-based speech recognition systems. Secondly, the AN based feature is a very promising technique and has the potential to replace the standard feature extraction methods once its two main limitations are addressed: the computing time and sensitivity to noise. Thus, working on solving these two issues can have a major impact on the field.

The findings related to adopting ESN when developing an Arabic speech recognition system are also discussed. The results, demonstrated in Chapter four, show the superior performance of the ESN approach compared to the recently-published models found in the literature that use the same corpora, to ensure a valid comparison, and the HMMs baseline models. This robust performance is achieved by models that can be trained and run on very modest machines (typical modern laptops) in less than an hour, using our approach for optimising the hyperparameters on a small reservoir size. This is a significant improvement compared to other recurrent neural network methods such as LSTM, where several GPUs are used to train the system for several days in order to achieve state-of-the-art performance. In addition, other recurrent and feed-forward neural network approaches need far more training samples to avoid over fitting issues. All of these attractive properties of the ESN prompted us to encourage society to adopt this approach in the Arabic speech recognition domain.

However, there are some limitations of the ESN, and the need to use a large reservoir size is the major weakness of this approach. In order to achieve the reported robust performance, a large reservoir size is needed and the model complexity in terms of the number of nodes grows as the square of the reservoir size which may result in hardware limitations issues, and RAM

issues are not unusual in the ESN model. In addition, the pseudo-inverse linear classifier that is typically used in the conventional ESN approach is very computationally expensive when a large corpus is used, which may lead also to hardware limitations. Thus, we have suggested working on these issues to reduce the needed complexity of the ESN model and increase its scalability. We also emphasise the importance of adopting the unsupervised learning paradigms to allow the ESN approach to make use of the huge and constantly growing unlabelled data available today for very low cost compared to labelled data. Obtaining this ability will have a major impact not only on reservoir-based speech recognition systems but also on many other fields. Thus, we highly recommend focusing on this topic. A more noise resilience ESN based system is also another promising area for future work.

Our novel approach (ESNSVMs) is also discussed in Chapter six, based on the results of the experiments presented in Chapter five, and we identify another promising research area for future work. Our findings suggest that ESNSVMs outperforms the ESN approach which means that a smaller reservoir based can be used to achieve a state-of-the-art competitive performance. This means that the ESNSVMs is more complex and has more parameters than the ESN approach when the same reservoir size is used. This however, is not an issue as ESNSVMs with a small reservoir size can outperform the ESN approach with a much larger reservoir size. Remembering that the overall complexity of the system decreases as the square of the reservoir size means that this reduction of the reservoir offered by the ESNSVMs, in many cases, leads to a signification decrease in the overall model complexity compared to ESN. The suggested future work areas include working on addressing the limitations of this approach such as the binary nature of SVMs to ensure that the approach can handle tasks involving a large number of different classes. In addition, applying this approach to a non-speech acoustic signal is a very promising area, based on the findings. Finally, adopting the developed

approach in applications that have large input dimensions such as computer vision tasks is another area for future work.

The second novel approach developed in this thesis is also discussed, namely ESNEKMs. The robust performance of the developed model compared to the ESNSVMs and ESN is emphasised. ESNEKMs maintains this superior performance even in the presence of different types and levels of noise and addresses the major limitation of the ESNSVMs by being able to handle multi-class tasks involving a large number of different classes. This has encouraged us to encourage the community to adopt this model for Arabic speech recognition systems. Future work includes implementing this model on continuous speech and integrating a language model to improve performance. This language model can be built using this model itself or by adopting a standard generative approach. In addition, applying this technique to a new domain such as handwriting classification is another promising area for future work. Developing a modified version that adopts the online learning paradigms enables this technique to be applied in far wider domains.

In addition, the effect of noise on the developed systems is discussed and we promote the use of novel pipelines, based on our findings. This architecture adopts the RASTA-PLP in the feature extraction method and ESNEKMs in the classification stage. We also highlighted the importance of the noise resilience approaches in order to ensure wide adoption in real-world applications. Future work includes investigating new feature extraction methods as the reservoir-based techniques enable the use of a far broader range due to its ability to handle large input dimensions, unlike the typical approaches.

Finally, the adopted, biologically-inspired feature extraction approach is discussed. The superior performance obtained by combining this approach with ESN on clean data has convinced us to consider the AN based feature to be a very promising approach. However, it shows poor performance in the presence of a high level of noise, particularly white noise, and the relatively

long computational time required to calculate the features create a serious barrier to its wider adoption. Thus, future work includes research on these two issues and tests the development approach on continues speech. In addition, applying this technique to other languages, such as English, will also be covered in future studies.

7.2 MEETING THE RESEARCH OBJECTIVES

The research objectives of this PhD study have been stated at the beginning of this thesis (see Chapter one for more details). In this section, we revisit these objectives and highlight how the work that has been presented in the previous chapters contributes to their accomplishment. The main objective of this study is to investigate the potential of adopting the reservoir computing approach to develop an Arabic speech recognition system. Several sub-objectives developed from this main objective. We will now consider each of them in turn and how this work addressed it.

- **To implement a reservoir computing-based Arabic speech recognition system**

  This has been presented in Chapter four, where we introduce the first reservoir-based Arabic speech recognition system.

- **To evaluate the performance of the developed system**

  The reservoir-based Arabic speech recognition system has been evaluated on three different corpora and the results are compared to those of other published studies and the HMMs baseline model.

- **To investigate the impact of the feature extraction methods on system performance**

To achieve this goal, the largest speech corpus for isolated words in the Arabic domain has been developed, due to the absence of the Arabic public corpus for isolated words that is accessible in a raw format. This developed corpus has been released to the public to ensure a high-level of reproducibility for our conducted research and allow other scholars to investigate new feature extraction methods. The results of this work are also presented in Chapter four.

- **To investigate the impact of the activation functions on system performance**

  The impact of the activation function on performance has been investigated via the ESN and ESNEKMs techniques using the Arabic phonemes corpus. Different feature extraction methods are considered to counter any feature extraction specific influence. These experiments are presented in Chapter five.

- **To develop a novel system based on reservoir computing**

  Two novel approaches have been developed and evaluated in this PhD study, namely ESNSVMs and ESNEKMs. These approaches are introduced and evaluated in Chapter five, while the research strengths and limitations are discussed in Chapter six. These two approaches have been introduced to the research community via two published papers.

- **To evaluate the developed system in the presence of noise**

  The ESN-based model and ESNEKMs have both been evaluated in the presence of different types and levels of noise. More than 10 different pipeline architectures have been considered under 7 different acoustic environments and each experiment has been repeated 10 times. This means that over 700 experiments have been conducted for this evaluation,

not including the training phase. The results of this work are reported in Chapter five.

Thus, we are confident that we have met our research objectives, given the imposed constraints. These constraints have been discussed in-depth in Chapter six and we have also identify the limitations of this PhD study and promising areas for future work that might build upon the work presented here.

7.3 FINAL WORDS

We have investigated the potential for adopting the reservoir computing approach in developing an Arabic speech recognition system. The first reservoir computing-based Arabic speech recognition system has been developed and compared to other state-of-the-art published models and the baseline model. The results of this evaluation process show the superior performance of the developed approach, which has been examined under different feature extraction methods, namely MFCCs, PLP, RASTA-PLP and the recently-developed novel biological inspired feature extraction method, the AN based feature. In order to implement these different architectures, we has developed the largest Arabic speech corpus of isolated words that contains about 10,000 samples. The proposed architectures have been evaluated under different levels and type of noises to investigate the performance of these systems in real-world applications where a high level of noise tends to be present.

In addition, we have built upon our findings and developed two novel approaches based on the reservoir computing approach; namely ESNSVMS and ESNEKMs. These two novel approaches have been evaluated on several corpora and compared to the state-of-the-art models found in the literature that use the same corpora, to ensure a valid comparison, as well as other

baseline models. The limitations and strengths of these models have been discussed in depth and we have also identified promising research areas for future work. Finally, we hope that this work will mark the beginning of the wider adoption of reservoir computing in this field and inspire the research community to focus on this promising topic.

BIBLIOGRAPHY

[1] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from data*. AMLBook, 2012.

[2] Abdulrahman Alalshekmubarak and Leslie S. Smith. A novel approach combining recurrent neural network and support vector machines for time series classification. In *Innovations in Information Technology (IIT) 9th International Conference on.*, pages pp. 42–47. IEEE, March 2013.

[3] Abdulrahman Alalshekmubarak and Leslie S. Smith. *On Improving the Classification Capability of Reservoir Computing for Arabic Speech Recognition*, pages 225–232. Artificial Neural Networks and Machine Learning (ICANN) 2014. Springer, 2014.

[4] Abdulrahman Alalshekmubarak and Leslie S. Smith. A noise robust arabic speech recognition system based on the echo state network. *The Journal of the Acoustical Society of America*, 135(4):2195–2195, 2014.

[5] Onsy Abdel Alim Ali, Mohamed M. Moselhy, and A. Bzeih. A comparative study of arabic speech recognition. In *Electrotechnical Conference (MELECON), 2012 16th IEEE Mediterranean*, pages 884–887. IEEE, 2012.

[6] YousefAjami Alotaibi. Comparing ann to hmm in implementing limited arabic vocabulary asr systems. *International Journal of Speech Technology*, 15(1):25–32, 2012. URL `http://dx.doi.org/10.1007/s10772-011-9107-3`.

[7] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2004.

[8] MA Anusuya and Shriniwas K. Katti. Speech recognition by machine, a review. *International Journal of Computer Science and Information Security*, 6(3), 2009.

[9] W. Astuti, AM Salma, AM Aibinu, R. Akmeliawati, and MJE Salami. Automatic arabic recognition system based on support vector machines (svms). In *National Postgraduate Conference (NPC), 2011*, pages 1–4. IEEE, 2011.

[10] J. Baker. The dragon system–an overview. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 23(1):24–29, 1975.

[11] Andrew G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

[12] Yoshua Bengio. *Practical recommendations for gradient-based training of deep architectures*, pages 437–478. Neural Networks: Tricks of the Trade. Springer, 2012.

[13] Christopher M. Bishop. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.

[14] George EP Box and Norman R. Draper. *Empirical model-building and response surfaces.* John Wiley & Sons, 1987.

[15] André R. Brodtkorb, Trond R. Hagen, Christian Schulz, and Geir Hasle. Gpu computing in discrete optimization. part i: Introduction to the gpu. *EURO Journal on Transportation and Logistics*, 2(1-2):129–157, 2013.

[16] Paulo R. Cavalin, Robert Sabourin, and Ching Y. Suen. Logid: An adaptive framework combining local and global incremental learning for dynamic selection of ensembles of hmms. *Pattern Recognition*, 45(9): 3544–3556, 9 2012.

[17] C. C. Chang and C. J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[18] Kyriakos C. Chatzidimitriou and Pericles A. Mitkas. A neat way for evolving echo state networks. In *ECAI*, pages 909–914, 2010.

[19] Noam Chomsky and Morris Halle. *The sound pattern of English.* New York: Harper & Row, 1968.

[20] CIA. *CIA Word Fact Book,Central Intelligence Agency,*, volume 2008. Central Intelligence Agency,Washington, D.C., Washington, D.C., 2008.

[21] Dan Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

[22] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[23] KH Davis, R. Biddulph, and S. Balashek. Automatic recognition of spoken digits. *The Journal of the Acoustical Society of America*, 24:637, 1952.

[24] Janez Demaiar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

[25] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, and Jason Williams. Recent advances in deep learning for speech research at microsoft. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8604–8608. IEEE, 2013.

[26] Thomas Dietterich. Ensemble methods in machine learning. *Multiple classifier systems*, pages 1–15, 2000.

[27] Daniel P. W. Ellis. Plp and rasta (and mfcc and inversion) in matlab, 2005. URL http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/.

[28] Jeffrey L. Elman. Finding structure in time. *Cognitive science*, 14(2): 179–211, 1990.

[29] S. Elmougy and AS Tolba. A comparison of combined classifier architectures for arabic speech recognition. In *Computer Engineering & Systems, 2008. (ICCES). International Conference on*, pages 139–153. IEEE Explorer, 2008.

[30] Ali Farghaly and Khaled Shaalan. Arabic natural language processing: Challenges and solutions. 8(4):14:1–14:22, dec 2009. URL http://doi.acm.org/10.1145/1644879.1644881.

[31] Aida A. Ferreira and Teresa Bernarda Ludermir. Genetic algorithm for reservoir computing optimization. In *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, pages 811–815. IEEE, 2009.

[32] Jonathan Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999.

[33] Sadaoki Furui. Fifty years of progress in speech and speaker recognition. *The Journal of the Acoustical Society of America*, 116(4):2497–2498, 2004.

[34] David Gamez. Progress in machine consciousness. *Consciousness and cognition*, 17(3):887–910, 2008.

[35] Ali Ganoun and Ibrahim Almerhag. Performance analysis of spoken arabic digits recognition techniques. *Power*, 2(1):0, 2012.

[36] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[37] Yifan Gong. Speech recognition in noisy environments: A survey. *Speech Communication*, 16(3):261–291, 1995.

[38] Alex Graves and Juergen Schmidhube. Offline handwriting recognition with multidimensional recurrent neural networks. In *In Advances in Neural Information Processing Systems*, pages 545–552, 2008.

[39] Z. Hachkar, A. Farchi, B. Mounir, and J. El Abbadi. A comparison of dhmm and dtw for isolated digits recognition system of arabic language. *International Journal of Computer Science and Engineering*, 3(3), 2011.

[40] Z. Hachkar, B. Mounir, A. Farchi, and J. El Abbadi. Comparison of mfcc and plp parameterization in pattern recognition of arabic alphabet speech. *Canadian Journal on Artificial Intelligence, Machine Learning & Pattern Recognition*, 2(3):56–60, 2011.

[41] N. Hammami and M. Bedda. Improved tree model for arabic speech recognition. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 5, pages 521–526, 2010.

[42] N. Hammami and M. Sellam. Tree distribution classifier for automatic spoken arabic digit recognition. In *Internet Technology and Secured Transactions, 2009. ICITST 2009. International Conference for*, pages 1–4, 2009.

[43] N. Hammami, M. Bedda, and F. Nadir. The second-order derivatives of mfcc for improving spoken arabic digits recognition using tree distributions approximation model and hmms. In *Communications and Information Technology (ICCIT), 2012 International Conference on*, pages 1–5, 2012.

[44] N. Hammami, M. Bedda, and F. Nadir. Probabilistic classification based on copula for speech recognitation: an overview. In *Computer Applications Technology (ICCAT), 2013 International Conference on*, pages 1–3, 2013.

[45] Dan Hammerstrom. Working with neural networks. *Spectrum, IEEE*, 30 (7):46–53, 1993.

[46] Bruce Hayes, Susan Curtiss, Anna Szabolcsi, Tim Stowell, Edward Stabler, Dominique Sportiche, Hilda Koopman, Patricia Keating, Pamela Munro, and Nina Hyams. Linguistics: An introduction to linguistic theory. *Malden, MA: Wiley-Blackwell*, 6:17, 2001.

[47] Hynek Hermansky. Perceptual linear predictive (plp) analysis of speech. *The Journal of the Acoustical Society of America*, 87:1738, 1990.

[48] Hynek Hermansky and Nelson Morgan. Rasta processing of speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578–589, 1994.

[49] Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, and Tara N. Sainath. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012.

[50] Nadia Hmad and Tony Allen. Echo state networks for arabic phoneme recognition. *World Academy of Science, Engineering and Technology International Journal of Computer, Information, Systems and Control Engineering*, Vol:7, No:7, 2013.

[51] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[52] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2): 415–425, 2002.

[53] Xiaohui Hu, Lvjun Zhan, Yun Xue, Weixing Zhou, and Liangjun Zhang. Spoken arabic digits recognition based on wavelet neural networks. In

Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on, pages 1481–1485. IEEE, 2011.

[54] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 2, pages 985–990. IEEE, 2004.

[55] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1):489–501, 2006.

[56] Guang-Bin Huang, Dian Hui Wang, and Yuan Lan. Extreme learning machines: a survey. *International Journal of Machine Learning and Cybernetics*, 2(2):107–122, 2011.

[57] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. Extreme learning machine for regression and multiclass classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 42 (2):513–529, 2012.

[58] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. Extreme learning machines, 2014. URL `http://www.ntu.edu.sg/home/egbhuang/elm_codes.html`.

[59] H. Jaeger. The" echo state" approach to analysing and training recurrent neural networks-with an erratum note'. *Tecnical report GMD report*, 148, 2001.

[60] Biing-Hwang Juang and Lawrence R. Rabiner. Hidden markov models for speech recognition. *Technometrics*, 33(3):251–272, 1991.

[61] Dan Jurafsky, James H. Martin, Andrew Kehler, Keith Vander Linden, and Nigel Ward. *Speech and language processing: An introduction to nat-*

ural language processing, computational linguistics, and speech recognition, volume 2. MIT Press, 2000.

[62] Manish P. Kesarkar. Feature extraction for speech recognition. In *Tech. Credit Seminar Report, Electronic Systems Group, EE. Dept, IIT Bombay*, 2003.

[63] L. I. Kuncheva. *Combining pattern classifiers: methods and algorithms*. Hoboken, NJ, J. Wiley, 2004.

[64] Langsci.ucl.ac.uk. Ipa: International phonetic association, 2014. URL `https://www.langsci.ucl.ac.uk/ipa/index.html`.

[65] K. F Lee. Context-dependent phonetic hidden markov models for speaker-independent continuous speech recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(4):599–609, 1990.

[66] Lee Lee-Min. Hmm speech recognition in matlab, Sep 21 2013. URL `http://sourceforge.net/projects/hmm-asr-matlab/`.

[67] Shane Legg and Marcus Hutter. Universal intelligence: A definition of machine intelligence. *Minds and Machines*, 17(4):391–444, 2007.

[68] Richard P. Lippmann. Review of neural networks for speech recognition. *Neural computation*, 1(1):1–38, 1989.

[69] M. Lukoševičius. A practical guide to applying echo state networks. *Neural Networks: Tricks of the Trade*, pages 659–686, 2012.

[70] Mantas Lukoševičius, Herbert Jaeger, and Benjamin Schrauwen. Reservoir computing trends. *KI-Künstliche Intelligenz*, pages 1–7, 2012.

[71] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the ICML*, 2013.

[72] W. Maass, T. Natschläger, and H. Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, 14(11):2531–2560, 2002.

[73] Angel Mario Castro Martinez, Niko Moritz, and Bernd T. Meyer. Should deep neural nets have ears? the role of auditory features in deep learning approaches. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[74] WarrenS McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of mathematical biophysics*, 5(4): 115–133, 12/01 1943. URL http://dx.doi.org/10.1007/BF02478259. J2: Bulletin of Mathematical Biophysics.

[75] Jonathan Milgram, Mohamed Cheriet, and Robert Sabourin. One against one or one against all : Which one is better for handwriting recognition with svms? In Guy Lorette, editor, *Tenth International Workshop on Frontiers in Handwriting Recognition*. Ecole de Technologie Superieure - ETS, Suvisoft, 2006-10-23 2006.

[76] M. Minsky and S. Papert. *Perceptrons; an introduction to computational geometry.* Cambridge, MA: MIT Press, 1969.

[77] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2012.

[78] Michael J. Newton and Leslie S. Smith. A neurally inspired musical instrument classification system based upon the sound onset. *The Journal of the Acoustical Society of America*, 131(6):4785–4798, 2012.

[79] Steven J. Nowlan and Geoffrey E. Hinton. Simplifying neural networks by soft weight-sharing. *Neural computation*, 4(4):473–493, 1992.

[80] J. Ross Quinlan. The effect of noise on concept learning. *Machine learning: An artificial intelligence approach*, 2:149–166, 1986.

[81] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[82] Abdel rahman Mohamed, George Dahl, and Geoffrey Hinton. Deep belief networks for phone recognition. In *NIPS Workshop on Deep Learning for Speech Recognition and Related Applications*, 2009.

[83] Tony Robinson, Mike Hochberg, and Steve Renals. *The use of recurrent neural networks in continuous speech recognition*, pages 233–258. Automatic speech and speaker recognition. Springer, 1996.

[84] Frank Rosenblatt. The perceptron-a perceiving and recognizing automaton. *Cornell Aeronautical Laboratory*, pages 85–460, 1957.

[85] David E. Rumelhart, Geoffrey E. Hintont, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[86] Leslie S. Smith and Andrew K. Abel. Spectrotemporal gabor filters for feature detection. *The Journal of the Acoustical Society of America*, 135(4): 2297–2297, 2014.

[87] Leslie S. Smith and Dagmar S. Fraser. Robust sound onset detection using leaky integrate-and-fire neurons with depressing synapses. *Neural Networks, IEEE Transactions on*, 15(5):1125–1134, 2004.

[88] Nitish Srivastava. *Improving neural networks with dropout*. PhD thesis, University of Toronto, 2013.

[89] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[90] F. Triefenbach and J. P Martens. Can non-linear readout nodes enhance the performance of reservoir-based speech recognizers? In *Informatics and Computational Intelligence (ICI), 2011 First International Conference on*, pages 262–267, 2011.

[91] Fabian Triefenbach, Kris Demuynck, and Jean-Pierre Martens. Improving large vocabulary continuous speech recognition by combining gmm-based and reservoir-based acoustic modeling. In *IEEE Workshop on Spoken Language Technology*, pages 107–112. IEEE, 2012.

[92] Fabian Triefenbach, Kris Demuynck, and Jean-Pierre Martens. Large vocabulary continuous speech recognition with reservoir-based acoustic models. *IEEE Signal Processing Letters*, 21(3):311–315, 2014.

[93] V. N. Vapnik. An overview of statistical learning theory. *Neural Networks, IEEE Transactions on*, 10(5):988–999, 1999.

[94] D. Verstraeten, B. Schrauwen, and D. Stroobandt. Reservoir-based techniques for speech recognition. In *Neural Networks, 2006. IJCNN '06. International Joint Conference on*, pages 1050–1053, 2006.

[95] David Verstraeten. *Reservoir Computing: computation with dynamical systems*. PhD thesis, Ghent University, 2009.

[96] Alex Waibel, Toshiyuki Hanazawa, Geoffrey Hinton, Kiyohiro Shikano, and Kevin J. Lang. Phoneme recognition using time-delay neural networks. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(3): 328–339, 1989.

[97] Grzegorz M. Wojcik and Wieslaw A. Kaminski. Liquid state machine built of hodgkin–huxley neurons and pattern recognition. *Neurocomputing*, 58:245–251, 2004.

[98] David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *Evolutionary Computation, IEEE Transactions on*, 1(1):67–82, 1997.

[99] X. Yang, K. Wang, and S. A. Shamma. Auditory representations of acoustic signals. *Information Theory, IEEE Transactions on*, 38(2):824–839, 1992.

[100] V. W. Zue. The use of speech knowledge in automatic speech recognition. *Proceedings of the IEEE*, 73(11):1602–1615, 1985.