IntechOpen

Computational Semantics

Edited by George Dekoulis and Jainath Yadav

Computational Semantics

Edited by George Dekoulis
and Jainath Yadav

Published in London, United Kingdom

Computational Semantics
http://dx.doi.org/10.5772/intechopen.102278
Edited by George Dekoulis and Jainath Yadav

Contributors
Eugene Fedorov, Tetyana Yuriyvna Utkina, Tetyana Neskorodieva, Jabbar Salman Hussein, Stephen A. Zahorian, Roozbeh Sadeghian, Xiaoyu Liu, Leonor Scliar-Cabral, Mohammed Abujoodeh, Liana Tamimi, Radwan Tahboub, Cristian Bosch, Ricardo Simon-Carbajo, George Dekoulis

First published in London, United Kingdom, 2023 by IntechOpen
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, 5 Princes Gate Court, London, SW7 2QJ, United Kingdom

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Additional hard and PDF copies can be obtained from orders@intechopen.com

Computational Semantics
Edited by George Dekoulis and Jainath Yadav
p. cm.
Print ISBN 978-1-83768-465-6
Online ISBN 978-1-83768-466-3
eBook (PDF) ISBN 978-1-83768-467-0

Meet the editors

Prof. George Dekoulis received his Ph.D. in Space Computing and Communications from the Computing and Communications Department, Lancaster University, UK, in 2007. He was awarded a First-Class Distinctions BEng (Hons) in Communications Engineering from the Faculty of Computing, Engineering and Media, De Montfort University, UK, in 2001. He has collaborated with all major space centers, including NASA and ESA. He is currently a member of the Faculty of Letters, University of Cyprus (UCY), Nicosia. He was previously the head of university research and founding dean of the Faculty of Sciences and Technology, American University of Cyprus (AUCY), Larnaca. His research focuses on the design of reconfigurable computing systems.

Dr. Jainath Yadav obtained an MTech and Ph.D. from the Indian Institute of Technology Kharagpur. He is currently an associate professor at the Central University of South Bihar, India. He has published several research papers in referred journals and presented several papers at international conferences. He is a member of the Institute of Electrical and Electronics Engineers (IEEE) and a Ph.D. supervisor in his active research areas.

Contents

Preface

This edited volume is a collection of reviewed research chapters on recent developments in computational semantics. It is divided into four sections: "Introduction", "Linguistics", "Classical Studies", and "Semantic Analysis in Computing".

After the "Introduction," the second section, "Linguistics", provides an overview of current speech recognition systems. It includes the following chapters: "Speech Recognition Based on Statistical Features", "Methods for Speech Signal Structuring and Extracting Features", and "Generalized Spectral-Temporal Features for Representing Speech Information".

The next section on "Classical Studies" includes one chapter: "Perspective Chapter: Difficulties for Translating Quevedo's Sonnets from Portuguese Translations into English".

The last section of the book, "Semantic Analysis in Computing", includes two contributions: "Toward Lightweight Cryptography: A Survey" and "Perspective Chapter: Computation of Wind Turbine Power Generation, Anomaly Detection and Predictive Maintenance".

We hope that you will enjoy reading this book and be inspired to scientifically contribute to the further success of the global computational semantics community.

George Dekoulis
Professor,
Department of Classics and Philosophy,
Faculty of Letters,
University of Cyprus,
Nicosia, Cyprus

Dr. Jainath Yadav
Department of Computer Science,
Central University of South Bihar,
Gaya, India

Section 1

Introduction

Chapter 1

Introductory Chapter: Introduction to Computational Semantics

George Dekoulis

> *"Some say that this is a sign of the soul, as it is buried in the present moment, and because through this, the soul signifies whatever it signifies, and this sign is rightly called a symbol."*
>
> *Plato, Cratylus, 400 BCE*

> *"Sema some say semaphores the psyche's burial into the soma during the present life. Because the psyche in the current soma semaphores polysemy, the semantic semantics are being semaphored"*
>
> *Agaiarch Diocles, 2023 CE*

1. Introduction

Computational semantics refer to the advanced scientific tools used for processing natural languages and extract interesting conclusions regarding the different meanings included. The models of the different languages should be well-understood and adequately put into the simulation and programming context. A language can be classified into three broad areas, including: syntax/structure, semantics/meanings and, finally, pragmatics. Seriatim, the principle of meaning can further be analysed. The main tool for reaching valid results is the efficient implementation of the logic principles involved in the modelling and computation processes [1].

2. Natural language characteristics

It took thousands of years for the different languages to evolve. It is this one skill of developing symbols, languages and communicating with each other that separates us from the animals. We have reached a great state of mind where a main Hellenic alphabet and subsequent ones have been created [2]. This allows us to form words, sentences and compile complete texts for specific subjects. Humans can efficiently express their thoughts and communicate with each other.

A great set of new scientific fields have been created, such as linguistics, in order to encapsulate the evolution of any language. In this field of science, it is always important to determine the qualities of the subject under investigation. Chomsky suggested various parameters and methods that can be used for correctly classifying a language [3]. A strong limitation in the modelling [4] and programming [5] stages has always been the level of understanding of the people involved in the different phases. Especially in the recent years where our computational capability [6] has

reached previously unseen levels of processing power [7], human limitation is still the restricting factor. For the purpose of further discussion, we will assume that a specific language can be defined through a textbook, archives or a series of representative sentences.

Grammar is probably the first thing we should seek in the skills of the various speakers. Grammar is a set of rules that stipulate and span throughout the language. The correct usage of the grammatic rules determines the efficiency of the implemented algorithms. Grammar demonstrates the following characteristics: Phonology, Morphology, Syntax and Semantics. Phonology distinguishes between tiny sounds and their combination into langer phonetic complexes. Morphology is responsible for the creation of words. Syntax is concerned with how words produce sentences. Semantics correspond to the meanings of the words that form sentences according to the syntactic rules.

In the current publication, we are focusing more on the top-level of language processing. The different chapters start from the discussion on phonology or morphology and elevate to semantics. Phonology is not further discussed in the current chapter. Based on the lingual rules that each of the authors has implemented the overall accuracy of the implemented algorithms is seriatim evaluated.

3. Morphology

In general, alphabetic languages can be analysed into their three counterparts: syntactic rules, the meanings of things and pragmatics. Syntax is more concerned with the art of combining morphemes and words into larger entities, as in phrases and sentences. To understand the purpose of semantics an excellent grasp of the corresponding language is needed and its grammar and syntax. In many languages, although we have a great knowledge of its constituents, we know little about their pragmatics usage. Pragmatics refer to the thoughts of the language users and the sentences that are being formed and exchanged between them to achieve communication. In this book, we are taking into consideration both pieces of speech and text of the English and Portuguese languages. Thus, we are considering samples of both natural and formal languages.

4. Semantics

In ethical teaching and philosophy, the aim is to communicate with each other, to describe and determine the truth and to acquire all the necessary virtues for a successful life. However, every language has historically being used to also deceive people, for the private benefit of the few. It is noticeable that no matter the education level of the participants it is frequent that the participants do not converge to a single truth. Logic and reasoning are techniques that every user should be trained to use [8]. This minimises the deviation between natural and formal languages when it comes to correctly defining a meaning. Semantics is what both states of a language share between them.

The implementation of state-of-the-art digital logic systems for various applications is the expertise of the author [9]. Logic has been used to create Hellenic, the first alphabetic language in the world [2]. Logic has been used to derive all the Hellenic dialects. Based on Hellenic, the other European languages have been logically derived, such as Spanish, Latin, French, English etc. Logic has been used extensively by the wise scholars to minimise the deviation between formal and natural morphemes [10]. Throughout human history, elements from the principles found in the field of discrete

mathematics have been found in the language formation phases down to the last detail of formally defining a language [11]. It is the usage of logical tools, such as predicate or propositional, that has permitted this. These are techniques being used by the authors of this book. A great analysis of the English language in terms of its natural and formal aspects is presented in [12]. It is well-known by ancient Hellenic that all the logical methods can be used into producing an extraordinary formal language. This is evident in the works of Homer, Hesiod, Socrates, Plato, Aristoteles, Proclus, the great Latin authors and many others. All these tools are also being used today to assist the convergence of natural and formal languages. Computational semantics is expedited by these techniques.

5. Computational semantics

Natural and formal language calculation of semantics has been based on [13] for many decades. Advanced programming techniques have been built around logical reasoning. Functional modelling and implementation have been built primarily around Montague reasoning. We are calling this field of mathematical thinking Λ (lamda) calculus. Functional modelling is a great expertise for linguists, since it allows them to parameterize all language aspects. Experimenting with the syntax, semantics and pragmatics is a means of evaluating all the classical theories. Through modelling the linguist gets immediate results and provides feedback to the theories under test.

The algorithms we are working on for computing semantics manipulate two categories of data representations. The first is the realisation of the various semantics. We are also post-processing the results acquired. The initial models are being put into extensive testing and the parameters of the initial models are accordingly adjusted. It is through extensive feedback that we have managed to build the various data retrieval software for searching through archives, computer databases and building impressive internet searching software. The combination of advanced computer science and artificial intelligence tools greatly assists in designing the next generation of natural and formal language processing tools. For instance, Haskell has historically been used for achieving functional modelling. Prologue was one the first programming languages used to implement predicate reasoning and perform engineering modelling [14]. Prologue does not meet today's needs for computing semantics. Haskell incorporated Prologue and a lot of programmers used it in computational semantics [15]. All modern high-level programming languages and dedicated hardware, preferably reconfigurable, are recommended for implementing computational semantics [16].

6. Conclusion

Natural language processing has always been intriguing linguists. However, high-performance programming has only been viable over the recent 20 years. In this publication, new state-of-the-art results are presented in the areas of natural and formal language speech processing, linguistics, classical studies and computational semantics. We anticipate this book to be an asset to researchers and the younger generations will be motivated to pursue studies in the areas of computer science, artificial intelligence, logic, linguistics and classical studies.

Author details

George Dekoulis
Department of Classics and Philosophy, Faculty of Letters, University of Cyprus, Nicosia, Cyprus

*Address all correspondence to: dekoulis.george@ucy.ac.cy

IntechOpen

References

[1] Dekoulis G. Field Programmable Gate Array. London, UK: INTECH; 2017. ISBN 978-953-51-3208-0

[2] Babiniotis G. The Hellenic Alphabet. Athens, Hellas: Kentro Lexicologias; 2018. ISBN 978-960-95-8213-1

[3] Chomsky N. Syntactic Structures. The Hague/Paris: Mouton; 1957

[4] Dekoulis G. Field Programmable Gate Array (FPGAs) II. London, UK: INTECH; 2020. ISBN 978-1-83881-057-3

[5] Dekoulis G. Novel space exploration technique for analysing planetary atmospheres. In: Air Pollution Vanda Villanyi. London, UK: IntechOpen; 2010. DOI: 10.5772/10053

[6] Dekoulis G. Novel digital magnetometer for atmospheric and space studies (DIMAGORAS). In: Aeronautics and Astronautics Max Mulder. London, UK: IntechOpen; 2011. DOI: 10.5772/17326

[7] Dekoulis G, Honary F. Novel Low-Power Fluxgate Sensor Using a Macroscale Optimisation Technique for Space Physics Instrumentation. SPIE, Smart Sensors, Actuators, and MEMS III. 2007;**6589**:65890G-1-65890G-8

[8] Dekoulis G, Honary F. Novel sensor design methodology for measurements of the complex solar wind – Magnetospheric, ionospheric system. Journal of Microsystem Technologies. 2008;**14**(4-5):475-482

[9] Dekoulis G. Robotics. London, UK: INTECH; 2018. ISBN 978-953-51-3636-1

[10] Dekoulis G. Drones – applications. London, UK: INTECH; 2018. ISBN 978-953-51-5948-3

[11] Lukasiewicz J. Aristotle's Syllogistic from the Standpoint of Modern Formal Logic. Oxford: Clarendon Press; 1951

[12] Montague R. English as a Formal Language: Formal Philosophy. New Haven and London: Yale University Press; 1974. pp. 188-221

[13] Montague R. The proper treatment of quantification in ordinary English. Approaches to Natural Language. 1973;**1973**:220-243

[14] Allison L. An executable Prolog semantics. Algol Bulletin. 1983;(50):10-18

[15] Spivey JM, Seres S. Embedding Prolog in Haskell. Utrecht: Department of Computer Science, University of Utrecht; 1999

[16] Allison L. A prolog semantics. In: A Practical Introduction to Denotational Semantics: Cambridge Computer Science Text. Cambridge; 1987. pp. 102-116

Section 2

Linguistics

Chapter 2

Speech Recognition Based on Statistical Features

Jabbar Hussein

Abstract

The requisition of intelligent devices that might classify a vocalized utterance have been skippering utterance research. The challenging task with utterance recognition models given for the language nature whereby there're nope apparent limits among words, an acoustic start with ending are impacted through the neighboring words, also, with various talkers utterance: female/male, senior/young, low/loud utterance, read/spontaneous, fast/slow vocalizing proportion and the utterance sign could be influenced by ambient noise. Accordingly, utterance recognition was exceeding abound of such challenges. To avert particular problems, information steered statistical curriculum built on considerable amounts of vocalized data has been utilized. With this itemize, the aim is to reconnoiter creativity that has making these implements plausible. Utterance recognition and language comprehension have been two important reconnoitering antes thereupon has normally been logged nearer as matters with indicatively and audio vocal, whereby the domain for audio vocal data have stayed introduced as robust impact to the matter thru drib accomplishment. Hence, we amid about determinate methods to utterances and language manipulating, whereby a data around a talking sign and a language that it converses, adjoining thru valuable utilized of information, is established come from inherent recognition of utterance data thru an understandable math-statistical formality.

Keywords: speech recognition, statistical features, language model and automatic speech recognition (ASR), language modeling, neural network

1. Introduction

The objective of getting a machine to see fluidly spoken talk and react in a characteristic voice has been driving taking research for over 50 years. We are as yet not yet where machines dependably comprehend familiar speech, spoken by anybody, and in any acoustic climate. Disregarding the excess specialized issues that should be tackled, the fields of automatic speech recognition (ASR) and comprehension have made enormous advances and the innovation is presently promptly accessible and utilized on an everyday premise in various applications and administrations [1, 2]. This chapter targets exploring the innovation that has made these applications conceivable. Talking recognition and language understanding are two significant exploration pushes that have customarily been drawn closer as issues in semantics and acoustic phonetics, where

scope of acoustic-phonetic information has been presented as a powerful influence for the issue with astoundingly little achievement. Here, in any case, we center around measurable techniques for speeches and language handling, where the information about a talking signal and the language that it communicates, along with useful usage of the information, is created from genuine knowledge of speech information through an obvious mathematical-statistical formalism.

2. Language Modeling (LM)

With this part, we will reflect on the issue of building a semantic model from a bunch of model words and sentences within an etymological. Semantic models were at first settled for the issue of 'Speech Recognition' (SR); they stay assume a predominant part in current (SR) frameworks. They are additionally regularly utilized in other (NLP) utilizes. The element assessment techniques that were initially settled for etymological demonstration, as characterized in this section, are significant in numerous different conditions, like the tagging and analysis problems [3].

Our occupation is as per the following. Expect that we take a body, which is a gathering of the sentences in one linguistic. A few years we could have of composition from the 'Washington Post', or we could own an exceptionally huge quantity of original copies by using the web. Accepted this corpus, we might want to estimate the elements of an etymological model. A semantic model is a clear cut as follows. To begin with, we will depict (V) to stand the gathering for entirely words within the language. For instance, once structure the phonetic system concerning the English language, we could say:

$$V = \{\text{that, cat, funs, maxim, bays, man,}\} \tag{1}$$

For all intents and purposes, (V) can be very large: it could have nearly thousands of words and we expect (V) to be a restricted set. Where a language sentence is a preparation of words, as:

$$x_1, x_2, \ldots, x_n.$$

Here (n) is the number with the end goal that ($n \geq 1$), where we consume: $x_i \in V$ for $i \in \{1 \ldots (n - 1)\}$, and where we expect to be (x_n) is a particular symbol, HALT (we accept that HALT is certainly not a partner of V). We'll in no time see the reason why it is appropriate to expect that each sentence decorations in the HALT symbol.

So (V) will depict to become the gathering of entirely sentences within the language V: here, this is a non-limitless group, since the sentences can be of different dimensions.

We next, at that point, provide the following description:

Definition: (LM) An etymological system includes of the restricted group V, also, $p(x_1, \ldots x_n)$ toward such an extent [4]:

- With any-value of $(x_1, \ldots x_n)$ V, then, at that point, $p(x_1, \ldots x_n) \geq 0$
- Also,

$$\sum_{(x_1 \ldots x_n) \in V^+} 1 + p(x_1, x_2, \ldots, x_n)$$

Where $p(x_1, \ldots x_n)$ is a likelihood distribution for the (V) sentences. For example, a delineation of a terrible strategy to the instruction of a phonetic system from a preparation body, contemplate a succeeding characterize $c(x_1, \ldots x_n)$ to being the time amount, where the $(x_1, \ldots x_n)$ is acknowledged in our preparation body, also (N) to stand for the complete amount from sentences during the preparation body. We might then characterize:

$$p(x_1 \ldots x_n) = \frac{c(x_1 \ldots x_n)}{N}$$

This is, anyway, an exceptionally unfortunate system: in explicit it shalt dispense (0) likelihood to somewhat sentence that is not understood in a preparation body. Accordingly, it will neglect to rearrange sentences that poor person was acknowledged in a preparation data. A critical useful commitment of hereupon section shalt be is adduce approaches that upon in all actuality carry out streamline for sentences that aren't understood in our preparation information. Firstly look, an etymological demonstrating issue appears similar to somewhat unusual work, thus, why it to be thought of? There is a pair of causes [5]:

1. Semantic systems are truly important for an expansive assortment to be uses, a clearest maybe SR plus machine transformation. Within numerous implementations, it's entirely important to own a decent "past" dissemination p $(x_1, x_2, \ldots, x_n)$ above whichever sentences are/aren't possible for the language. For an instance, in SR the phonetic model is joint with an audio system that models the way to express different words: Certain strategy for consider it's that upon the audio system produces countless candidate sentences, created with probabilities; a semantic system is then used to rework these choices in light of the fact that they are so plausible to being the sentence within a language.

2. A strategies we are characterized to portraying a (p), then for speculating elements come from the resultant system for showing models, can stand for help with various settings all through the course; for instance, in Neural Network (NN) and Hidden Markov Models (HMM), that we shalt acknowledge subsequently, and for systems to the standard language depicting.

3. Statistical features

Every talking signal equivalent for some word is placed in an individual file. Various talking features can be considered, deeming the vocalized words just as an acoustic sign, and come from that the acoustic features could be elicited and so, generally categorized built onto their semantic clarification just as cognitive and physical traits. Furthermore, statistical traits containing, RMS, absolute mean value (AMV), median absolute value (MAV), standard deviation (STD), variance value, covariance, maximum & minimum values and others, as follows [6, 7]:

3.1 AMV

It's come from the outright measure for the sign information. Quite possibly the most standard component could be utilized during the features elicited. It's established by:

$$\overline{\mathrm{P}} = 1/\mathrm{R}\sum_{\mathrm{r}=1}^{\mathrm{R}} \mathrm{P_r} \tag{2}$$

Here (P_r) is the information vector, and (R) is the input vector size.

3.2 STD

The (STD) element can be accustomed to working out the value of mean-variation for every part in signal information. It is established by:

$$STD = \sqrt{\frac{1}{R-1}\sum_{r=1}^{R}\left(P_r - \overline{P}\right)^2} \tag{3}$$

3.3 Variance

It is the square of (STD). It is established by:

$$VAR = \frac{1}{R-1}\sum_{r=1}^{R}\left(P_r - \overline{P}\right)^2 \tag{4}$$

3.4 RMS

As a (MAV) of signals oftentimes will quite often way to be or nearly be zero, an RMS is a best gauging to the qualities of the signs. RMS will be predefined by means of a square-root for the sign mean-square. It will connect with (STD) and is characterized as:

$$RMS = \sqrt{\frac{1}{R}\sum_{r=1}^{R} {P_r}^2} \tag{5}$$

3.5 Maximum & minimum values

They could be deemed just as significant features for the sign. Where they could be founded through calculating the biggest and the tiniest amounts of these data, just as predefined in the subsequent:

$$P_{max} = \max\left(P_1, P_2, \ldots, P_R\right) \tag{6}$$

$$P_{min} = \min\left(P_1, P_2, \ldots, P_R\right) \tag{7}$$

3.6 MAV

MAV of signs could be founded by the calculating the medium amount in a group of progressives arranged absolute amounts. With two midpoint values, the medium shalt is the mean of those amounts.

4. Acoustic modeling and recognition methods

A worthy quality for (LM) is reflected to be a vital piece of a few frameworks for language information applications, like (SR), machine interpretation, and so on. The point of an LM is to characterize likely series of pre-defined language units, which are normally words. Syntactic and semantic and attributes of a language, coded through the LM, director these figures [8].

4.1 Neural network (NN)

The point of a semantic system is to rating the likelihood conveyance $p(w_1^T)$ for word- sequence (w_1^{t-1} = w1, … ,wT) . Through the 'chain norm', so, for this conveyance could be uttered as:

$$p(w_1^T) = \prod_{t=1}^{T} p(w_t|w_1^{t-1}) \tag{8}$$

Accompanying for a particular segment displays how Recurrent NN (RNN) and Feedforward NN (FNN) have been utilized for assess this particular likelihood conveyance [9].

4.1.1 FNN

Correspondingly with N-gram system, the FNN usages Markov-theory of tidiness: (N-1 to approximate 1) giving for:

$$\mathrm{p}(\mathrm{w}_1^{\mathrm{T}}) \approx \prod_{\mathrm{t}=1}^{\mathrm{T}} \mathrm{p}(\mathrm{w}_\mathrm{t}|\mathrm{w}_{\mathrm{t}-\mathrm{N}+1}^{\mathrm{t}-1}) \tag{9}$$

Consequently, each one with the terms convoluted within this creation, such as: $p(w_t|w_{t-N+1}^{t-1})$, was expected, distinctly, with one progressive estimation for a network depending on:

$$\mathrm{P}_{\mathrm{t}-\mathrm{j}} = \mathrm{X}_{\mathrm{t}-\mathrm{j}}.\mathrm{U}, \mathrm{j} = \mathrm{N}-1, \ldots, 1 \tag{10}$$

$$\mathrm{H}_\mathrm{t} = \mathrm{f}\left(\sum_{\mathrm{t}-1}^{\mathrm{N}-1} \mathrm{P}_{\mathrm{t}-\mathrm{j}}.\mathrm{V}_\mathrm{j}\right) \tag{11}$$

$$\mathrm{O}_\mathrm{t} = \mathrm{g}(\mathrm{H}_\mathrm{t}.\mathrm{W}) \tag{12}$$

Where (X_{t-i}) represents one coding for a word (w_{t-i}), while a (U) columns coding a continual word outline (i.e., embedding). Subsequently, (P_{t-i}) i represents a continual outlines for a (w_{t-i}) word. V = [V_1,V_2, … , V_{N-1}] and (W) were the system connecting weighs, where they are educated all through preparing adding (U). Besides, the function f(.) is an initiation work, while the function g(.) is the softmax one. **Figure 1**-a displays a representation of a FNN through an extremely durable setting magnitude (N-1 = 3) thru a hidden stratum equal one.

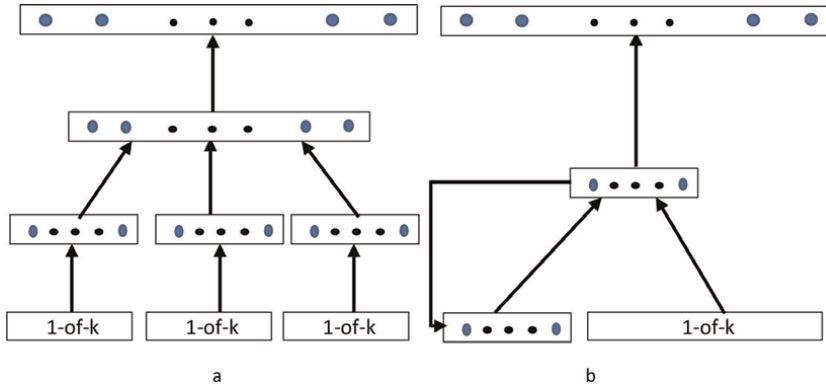

Figure 1
FNN vs. RNN architecture, a) FFNN and b) RNN.

4.1.2 RNN

An RNN endeavor for catching entire histories within the setting parameter (h_t), whichever implies a condition for a system and also advances on schedule. Thus, it approaches (2) providing for [10]:

$$p(w_1^T) = \prod_{i=1}^{T} p(w_i|w_{i-1}, h_{i-1}) = \prod_{i=1}^{T} p(w_i|h_i) \tag{13}$$

RNN calculates this particular proration correspondingly for FNN. A major variance happens within Eqs. (10) and (11) which they are joined to:

$$H_i = f(X_{i-1}.U + H_{i-1}.V) \tag{14}$$

Figure 1-b shows an illustration of a typical RNN.

4.2 Hidden markov model HHM

We now turn to an important question: given a training body, in what way do we training the function (p)? With section we define HMM, a dominant idea from probability theory [11].

4.2.1 Sustained-length series markov models

Deem a series for arbitrary parameters, such as: (X_1,X_2, … …,X_n). Every arbitrary parameter could offtake whichever amount during a limited group (V). Until now we shalt adopt thereupon the dimension for the series (n), will be some permanent integer (such as: n = 250).

Our point is as per the following: we might want to demonstrate the series likelihood (x_1,x_2, … x_n), here ($n \geq 1$),also, {$x_j \in V$ for (j = 1 … n)}, thereupon for to say, and show a combined likelihood.

$P(X_1 = x_1, \ldots..,X_n = x_n)$.Where $|V|^n$ possible series of the form x1 ... xn are there, thus obviously, it's not possible to the reasonable amounts for (|V| & n) being only listing whole ($|V|^n$) eventuality. Next, we shalt to show the HMM for the same case with the applications of features extracting and recognition.

4.2.2 HMM and one-state method

HMM is a random structure utilized to forecast a greeter event reliant depending on the preceding data. A structure contains a group of statuses, whereby merely an output for the statuses could be observed, and so, whole the variations between the statuses are unidentified as shown in **Figure 2**. The HMM could be clustered into two categories just as shown through a knowing of an outputs: discrete HMM (DHMM) and continues HMM (CHMM).

With Discrete DHMM, this kind achieves (discrete-codes) which are moved through the states and the design (λ) is laid out by the 3-limits (π, A, B).

While, with Continuous CHMM, "continuous" assigns the possibility of the result concentrations of the covered states. Comparable a Gaussian-limit, the results path the 'Probability Density Function (PDF)', here it's the symmetrical curve outlining the strategy looks like the ring. So, a discernment vector (O), the PDF is found through a second proviso:

$$P(O) = \sum_{n=1}^{k} \frac{w_n}{\sqrt{2\pi\sigma^2}} \exp\left[\frac{(O-\mu_n)^2}{\sigma_n^2}\right] \tag{15}$$

Here: w_n: the weight, σ_n: the standard deviation and μ_n: the mean for n^{th}-Gaussian mixture. It's significant thereupon vector covariance ($\sum$) is corresponding for a squared of the (σ_n) therefore, the CHMM is characterized as during the related group: Here: (w_n, μ_n and σ_n) are: independently, the weight, mean and standard deviation of the n^{th} Gaussian mix. It's critical thereupon a ($\sum$) will be a comparing for squared of (σ_n), so, in this way, the CHMM is described by means of the related group:

$$\lambda = (\mu, \Sigma, \pi, n, A) \tag{16}$$

Resulting focuses offer a synopsis of its design:

- N: Structure states number.
- M: Result code number.

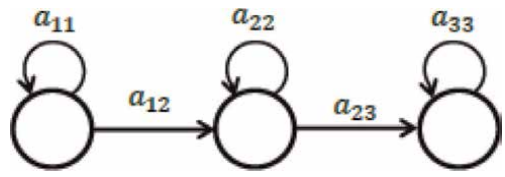

Figure 2.
Status graph for 3-status L-R HMM.

- π: A major status likelihood element size (N × 1).
- A: A variety likelihood design size (N × N).
- B: The delivery likelihood design size (N × M).

A distinction among a CHMMs & DHMMs, with respect to the HMM limits, is in dis-charge limit, where within CHMM; it's identified through a mean & covariance slightly from separate-codes.

4.2.3 Features elicited

During this deed, features elicited & classification were applied. Every speech sign equivalent for some word will be placed inside a particular file. Various talking features can be considered, furthermore, statistical features containing, RMS, AMV, MAV, STD, variance value, covariance, max. & min. Value and others.

During this effort, a statistical elicited: a mean value & covariance were the features utilized, since the statistical traits characterize a central for the sign and so decrease a necessary magnitude and the time of treating.

4.2.4 Recognition

During the recognition phase, the work is done through two portions:

A. Training phase and.

B. Testing phase, as follows:

4.2.4.1 Network training (NTr)

For each articulated word, and by joining all the series contracted from the (NTr) word, an array is formed. Whenever the array is outlined, it is given to the HMM to NT. Well applied in the work is an unmistakable framework thereupon comprises only one-status thru ceaseless result densities. No (π) and (A), happen during the one-status framework so, in the present circumstance, they are equal to one. Thusly, a framework (λ) is ordinarily established upon the (∑ and μ) for the advised vectors, just as displayed in **Figure 3**.

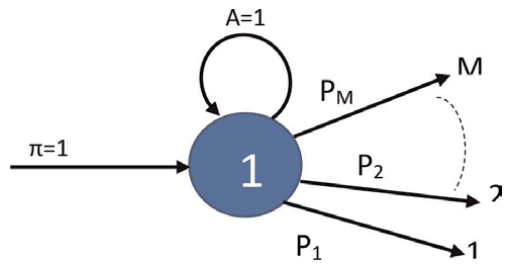

Figure 3.
State diagram of the continuous one-state model (COSM).

To prepare the word arrangement, a 'Baum-Welch' framework thru one-cycle was applied. Simply, by using single Gaussian-blend with (PDFs) were founded just as with Eq. (15), here $P(O) = [P_1,P_2,P_3,...,P_M]$. COSM status outline is shown in **Figure 2**.

4.2.4.2 Network testing (NTs)

For the network test, whole articulated words thereupon are not used during the HMM-NTr path, a comparable supra points, where every word will be individually handling. A ($\sum$ and μ) for discernment vectors were ascertaining and also the Viterbi computation was applied to get their eventualities through the whole (PDFs) where they are contracted during a preparation technique. Then, at that point, the file of the best outrageous likelihood may be applied to separate the new word.

4.2.4.3 Example study

Tests are achieved on the work information bases: here with 5-people, 100 examples for everyone, NTr with 70 words & NTs with 30 words. It was a difficult advance in this work for information generation and assortment, on the grounds that the Arabic words sound extremely infrequent on the Internet and furthermore, the works and exploration about verbally expressed the Arabic words are exceptionally inadequate. Thus, recording the audio of the Arabic words from people's lives nearby us were the strategies utilized. The recording system is completed by utilizing (BOYA BY-M1) amplifier. Likewise, a Matlab (2017) utilized as the program that the greater part of the work done through it. Mono-sound with 16-digit coding, 1-channel, and (8000 Hz) sampling frequency. That requirement is picked on since that, the size of each recorded word is vital, as the size is lesser, the method of all tasks follows is quicker and less memory utilized.

Through (HMM), the tests show that the strategy for utilizing the (μ and $\sum$) are the well one. Along these lines, this strategy is tried utilizing CHMM, and the accompanying particulars are worked:

1-Pre-processing: For the words information base: first phase of (DWT) give of 2002×1 vector size.

2- Covered Hamming window with 75% (overlapped) with (n = 100) of length.

3-Feature elicited: $C = [MV\ MN]$.

4- NTr.

Afterward an information assembly, so, we attempted our knowledge computation as assignments later:

- For arbitrary reasons choice (70)
- Test for the remainder (30)
- Playback phases (1 & 2) ordinarily

Here phase (c) will be changed up from the choice of the readiness set.

5. The results

The outcomes showed in **Table 1**, are laid out for 5-people every one has 100-words, preparing with 70 and the testing comes with 30.

Speakers	Training words	Test words	Recognition Rate %
1	70	30	100
2	70	30	100
3	70	30	100
4	70	30	100
5	70	30	100

Table 1
Recognition ratio for HMM.

Patterns for each individual are taken, as shown in **Figure 4**, furthermore launched thru 70 one to the NTr, with 30 for NTs, now, at that point, with the phase of 5-words, we will diminished preparation patterns with a test one expanded, our expectation for find the impact for a quantity of patterns thereto HMM calculation with a recognition ratio, just as displayed with **Figure 5**, a recognition ratio diminished according to diminishing the preparation patterns, so, thereupon is the standard outcome according to such calculation.

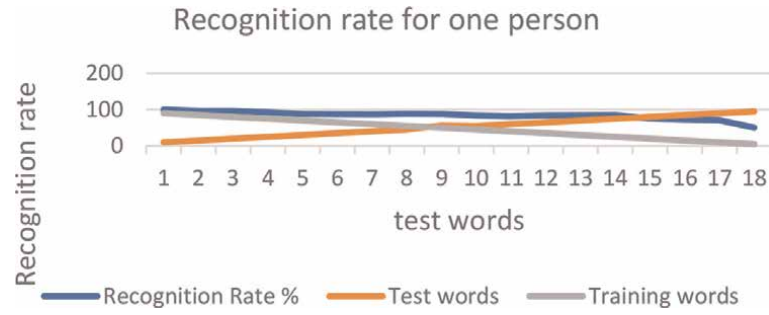

Figure 4.
One person recognition ratio with variables (NTr & NTs) words.

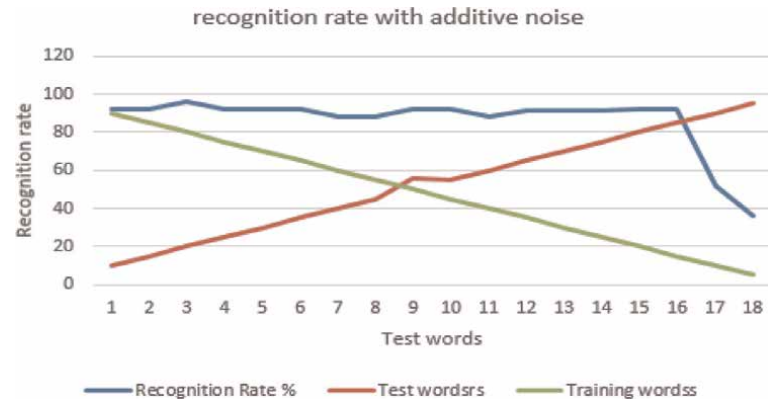

Figure 5.
One person recognition rate with additive noise.

Speakers	Training words	Test words	Recognition Rate %
1	70	30	91.6
2	70	30	83.3
3	70	30	91.6
4	70	30	83.3
5	70	30	83.3

Table 2
Recognition rate for HMM with AWGN.

Speakers	Training words	Test words	Recognition Rate % One state HMM	Recognition Rate % MLFFNN
1	70	30	100	90
2	70	30	100	90
3	70	30	100	90
4	70	30	100	90
5	70	30	100	90

Table 3
HMM comparison with MLFFNN.

To mimic the impacts of noise or fault with a presentation for a recognition framework, an 'Additive White Gaussian Noise' (AWGN) is strengthening for a patterns samples, preparing and test, since like the clamor cavers whole range, an outcomes display great results, just as displayed with **Table 2**. Anyway, with **Figure 2**, it displays an AWGN impact regard one-individual recognition. While with less commotion values, the results could be improved.

To make a comparison thru different methods, as NN, as Feed Forward NN (FFNN), as displayed in **Table 3**, one can see that the HMM has better outcomes.

6. Conclusions

With SR, it means the usage of an intelligent machine for recognizing spoken word. SR models could be utilized to recognize certain word or to verify a spoken word. Talking processing, talking production, features elicited and finally, patterns equivalent to the SR were presented. Our work has been led us to conclude that the statistical features of the signal are over-performing than the physical features of that signal. The preprocessing step is important for the classification goal.

Author details

Jabbar Hussein
Collage of Engineering, Kerbala University, Kerbala, Iraq

*Address all correspondence to: jabbar.salman@uofkerbala.edu.iq

IntechOpen

References

[1] Raut PC, Deoghare SU. Automatic speech recognition and its applications. International Research Journal of Engineering and Technology (IRJET). 2016;**03**(05):2368

[2] Wiqas G, Navdeep S. Literature review on automatic speech recognition. International Journal of Computer Applications. 2012;**41**(8):0975-8887

[3] Rafal J, Oriol V, Mike S, Noam S, Yonghui W. Exploring the limits of language modeling, Google brain. arXiv: 1602.02410v2 [cs.CL]. 11 Feb 2016;**2**

[4] Statistical Speech Recognition: A Tutorial, MC_He_Ch02.Indd. Achorn International; 2008

[5] Michael C. Language Modeling, (Course Notes for NLP, Columbia University, Columbia). lm-spring; 2013. Available from: http://www.cs.columbia.edu/~mcollins/lm-spring2013.pdf

[6] Othman OK, Khalid K, Aisha HA, Jamal ID. Statistical modeling for speech recognition. World Applied Sciences Journal 20 (Mathematical Applications in Engineering). IDOSI Publications. 2012;**20**:115-122. DOI: 10.5829/idosi.wasj.2012.20.mae.99935. ISSN: 1818-4952

[7] Husam A, Hala BAW, Abdul MJ, A. H. A new proposed statistical feature extraction method in speech emotion recognition. Computers & Electrical Engineering. 2021;**93**:107172

[8] Youssef O, Dietrich K. A neural network approach for mixing language models. arXiv:1708.06989v1 [cs.CL]. 23 Aug, 2017;**1**

[9] Sundermeyer M, Oparin I, Gauvain JL, Freiberg B, Schluter R, Ney H. Comparison of Feedforward and Recurrent Neural Network Language Models, 978–1–4799-0356-6/13. IEEE. 8430 ICASSP; 2013

[10] Youssef O, Clayton G, Mittul S, Dietrich K. Sequential recurrent neural networks for language modeling. arXiv: 1703.08068v1 [cs.CL]. 23 Mar, 2017;**1**

[11] Jabbar SH, Abdulkadhim AS, Thmer RS. Arabic speaker recognition using HMM. Indonesian Journal of Electrical Engineering and Computer Science. 2021;**23**(2):1212-1218. ISSN: 2502-4752, DOI: 10.11591/ijeecs.v23.i2.pp 1212-1218

Chapter 3

Methods for Speech Signal Structuring and Extracting Features

Eugene Fedorov, Tetyana Utkina and Tetiana Neskorodieva

Abstract

The preliminary stage of the biometric identification is speech signal structuring and extracting features. For calculation of the fundamental tone are considered and in number investigated the following methods – autocorrelation function (ACF) method, average magnitude difference function (AMDF) method, simplified inverse filter transformation (SIFT) method, method on a basis a wavelet analysis, method based on the cepstral analysis, harmonic product spectrum (HPS) method. For speech signal extracting features are considered and in number investigated the following methods – the digital bandpass filters bank; spectral analysis; homomorphic processing; linear predictive coding. This methods make it possible to extract linear prediction coefficients (LPC), reflection coefficients (RC), linear prediction cepstral coefficients (LPCC), log area ratio (LAR) coefficients, mel-frequency cepstral coefficients (MFCC), barkfrequency cepstral coefficients (BFCC), perceptual linear prediction coefficients (PLPC), perceptual reflection coefficients (PRC), perceptual linear prediction cepstral coefficients (PLPCC), perceptual log area ratio (PLAR) coefficients, reconsidered perceptual linear prediction coefficients (RPLPC), reconsidered perceptual reflection coefficients (RPRC), reconsidered perceptual linear prediction cepstral coefficients (RPLPCC), reconsidered perceptual log area ratio (RPLAR) coefficients. The largest probability of identification (equal 0.98) and the smallest number of coefficients (4 coefficients) are provided by coding of a vocal of the speech sound from the TIMIT based on PRC.

Keywords: speech recognition, speech signal structuring and extracting features, the digital bandpass filters bank, spectral analysis, homomorphic processing, linear predictive coding

1. Introduction

Most often from a speech signal the following features are distinguished [1–10]: power features (energy of a spectral bands); cepstrum; linear predictive parameters; fundamental tone and formant; mel-frequency cepstral coefficients (MFCC); bark-frequency cepstral coefficients (BFCC); parameters of perceptual linear prediction; parameters of the reconsidered perceptual linear prediction.

For features extraction of a speech signal usually use [1–10]: digital bandpass filters bank; spectral analysis (Fourier's transformation, wavelet transformation); homomorphic processing; linear predictive coding; MFCC method; BFCC method; perceptual linear prediction; reconsidered perceptual linear prediction.

2. Calculation methods of the fundamental tone

For calculation of the fundamental tone use methods which are based on a basis of the analysis of the following signal representations [3]: amplitude-time; spectral (amplitude-frequency); cepstral (maplitude-quefrency); wavelet-spectral (amplitude-time-frequency).

2.1 ACF method

The autocorrelation function (ACF) method carries out search of the maximum value in autocorrelated function [3]:

1. For the chosen signal frame of length ΔN calculates autocorrelated function

$$R(k) = \frac{1}{\Delta N} \sum_{n=0}^{\Delta N-1-k} x(n)x(n+k), \quad k \in \overline{0, \Delta N - 1}. \tag{1}$$

2. Impulse response function initialization Is defined at what value k autocorrelated function $R(k)$ it is maximum that corresponds to extraction of the periods in a speech signal

$$k^* = \arg\max_k R(k), \quad k \in \overline{0, \Delta N - 1}. \tag{2}$$

The period of the fundamental tone is defined in a form

$$T_{\text{OT}} = \begin{cases} k^*, & k^* \in [n_2, n_2] \\ 0, & k^* \notin [n_2, n_2] \end{cases}, \tag{3}$$

where n_1—minimum length of the fundamental tone period, $n_1 = \inf T_{\text{OT}}$, n_2—maximum length of the fundamental tone period, $n_2 = \sup T_{\text{OT}}$.

2.2 AMDF method

The average magnitude difference function (AMDF) method carries out search of the minimum value as the average magnitude difference [3] that quicker than search of the maximum value in autocorrelated function.

1. For the chosen signal frame of length ΔN calculates function of the average magnitude difference

$$v(k) = \frac{1}{\Delta N} \sum_{n=0}^{\Delta N-1} |x(n) - x(n+k)|, \quad k \in \overline{0, \Delta N - 1}. \tag{4}$$

2. Is defined at what value k function of the average magnitude difference $v(k)$ it is minimum that corresponds to extract of the periods in a speech signal

$$k^* = \arg\min_k v(k), \quad k \in \overline{0, \Delta N - 1}. \tag{5}$$

The period of the fundamental tone is defined in a look

$$T_{\text{OT}} = \begin{cases} k^*, & k^* \in [n_2, n_2] \\ 0, & k^* \notin [n_2, n_2], \end{cases} \tag{6}$$

where n_1—minimum length of the period of the fundamental tone, $n_1 = \inf T_{\text{OT}}$, n_2—maximum length of the period of the fundamental tone, $n_2 = \sup T_{\text{OT}}$.

2.3 SIFT method

The simplified inverse filter transformation (SIFT) method carries out search of the maximum value in autocorrelated function of linear prediction error of the decimated signal [4]:

1. For the chosen signal frame of length ΔN extracted the frequency range containing the frequency of the fundamental tone by means of elliptic LPF with a cut frequency f_{cut}=1000 Hz. Instead of the elliptic LPF used in [4] the consecutive calculation is offered:

   - DFT (discrete Fourier transform)

$$X(k) = \sum_{n=0}^{\Delta N-1} x(n) e^{-j(2\pi/\Delta N)nk}, \quad k \in \overline{0, \Delta N - 1}; \tag{7}$$

   - extract of the lower frequencies

$$X_{low}(k) = \begin{cases} X(k), & 0 \le k \le k_{cut} \\ 0, & k_{cut} < k \le \Delta N - 1 \end{cases}, \quad k_{cut} = \left[f_{cut} \cdot \Delta N / f_d \right], \tag{8}$$

     where f_d—sampling frequency;

   - calculation of the inverse DFT

$$y(n) = \text{Re}\left(\frac{1}{\Delta N} \sum_{k=0}^{\Delta N-1} X_{low}(k) e^{j(2\pi/\Delta N)nk} \right), \quad n \in \overline{0, \Delta N - 1}. \tag{9}$$

2. Decreases sampling frequencies to $f1_d$=2000 Hz by decimation of a signal, i.e. are removed intermediate samples of a signal

$$s(n) = y(n \cdot \Delta n), \quad n \in \overline{0, \Delta N / \Delta n - 1}, \tag{10}$$

where $\Delta n = \left[f_d / f1_d \right]$—decimation coefficient, f_d—sampling frequency.

3. The differences of two next samples of the decimated signal are calculated

$$s_{\Delta}(n) = \begin{cases} s(n), & n = 0 \\ s(n) - s(n-1), & n > 0 \end{cases}, \quad n \in \overline{0, \Delta N/\Delta n - 1}. \tag{11}$$

4. Autocorrelated function is calculated

$$\hat{s}_{\Delta}(n) = \breve{s}_{\Delta}(n) w(n), w(n) = 0.54 + 0.46 \cos \frac{2\pi n}{\Delta N}, \tag{12}$$

$$R(k) = \sum_{n=0}^{\Delta N/\Delta n - 1 - k} \hat{s}_{\Delta}(n) \hat{s}_{\Delta}(n+k), \quad k \in \overline{0, p}, \tag{13}$$

where $w(m)$—Hamming's window, p—order of linear prediction, $ceil(f1_d/1000) \le p \le 5 + ceil(f1_d/1000)$, $ceil(f)$—function which rounds f to the next integer.

5. LPC coefficients are calculated a_j according to the procedure Darbin.

6. The error of linear prediction by means of LPC coefficients is calculated

$$e(n) = \begin{cases} s(n), & n < p \\ s(n) - \sum_{k=1}^{p} a_k s(n-k), & n \ge p \end{cases}, \quad n \in \overline{0, \Delta N/\Delta n - 1}, \tag{14}$$

where $e(n)$—prediction error.

7. Autocorrelated function of a linear error of prediction is calculated

$$e_w(n) = e(n) w(n), w(n) = 0.54 + 0.46 \cos \frac{2\pi n}{\Delta N}, \tag{15}$$

$$r(k) = \sum_{n=0}^{\Delta N/\Delta n - 1 - k} e_w(n) e_w(n+k), \quad k \in \overline{0, \Delta N/\Delta n - 1}, \tag{16}$$

where $w(m)$—Hamming's window.

8. Is defined at what value k autocorrelated function $r(k)$ it is maximum that corresponds to extraction of the periods in a speech signal

$$k^* = \arg\max_k r(k), r^* = \max_k r(k), \quad k\Delta n \in [n_1, n_2], \tag{17}$$

where n_1—minimum length of the fundamental tone period, $n_1 = \inf T_{\text{OT}}$, n_2—maximum length of the fundamental tone period, $n_2 = \sup T_{\text{OT}}$. Thus, length of the fundamental tone period is determined in a form

$$T_{\text{OT}} = \begin{cases} k^* \Delta n, & r^* \ge \gamma \\ 0, & r^* < \gamma \end{cases}, \tag{18}$$

where γ—the threshold value.

Example 1

In **Figure 1** the source signal, is presented on **Figure 2**—noisy (additive white is added the noise with a mean 0 and variance 0.001 is Gaussian), on **Figure 3**—filtered and $M = 1$.

As a signal the frame of a sound "A" length is chosen ΔN= 512 with a sampling frequency f_d=22050 Hz, 8 bits, mono. In **Figures 1–6** the initial signal (**Figure 1**), the filtered signal (**Figure 2**), the decimated signal (**Figure 3**), a signal in the form of the weighed difference (**Figure 4**), an error of prediction (**Figure 5**), autocorrelated function of an error of the prediction with extraction of the found maximum and admissible boundaries (**Figure 6**) are presented.

2.4 Method on a basis a wavelet analysis

This method calculates distance between the next minimum a wavelet coefficients.

At the first stage the continuous wavelet transformation which is approximated according to a rectangles formula in a look is calculated

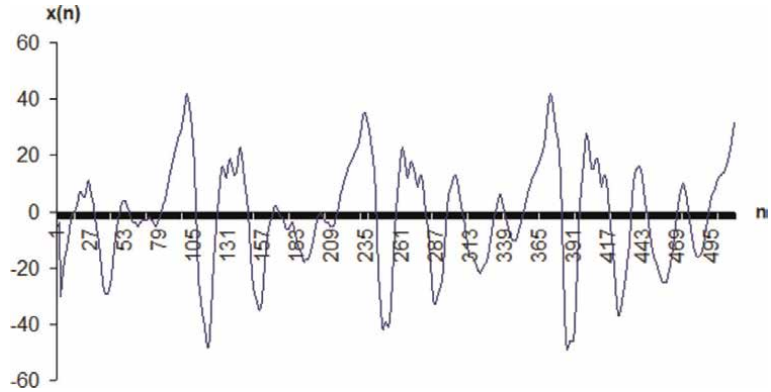

Figure 1.
Initial signal.

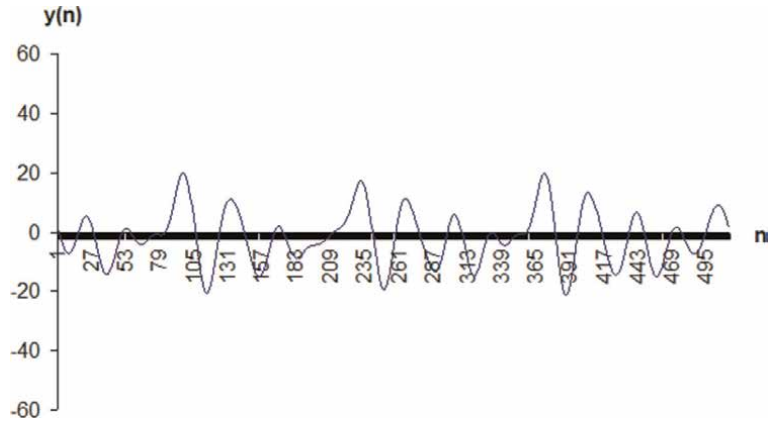

Figure 2.
The filtered signal.

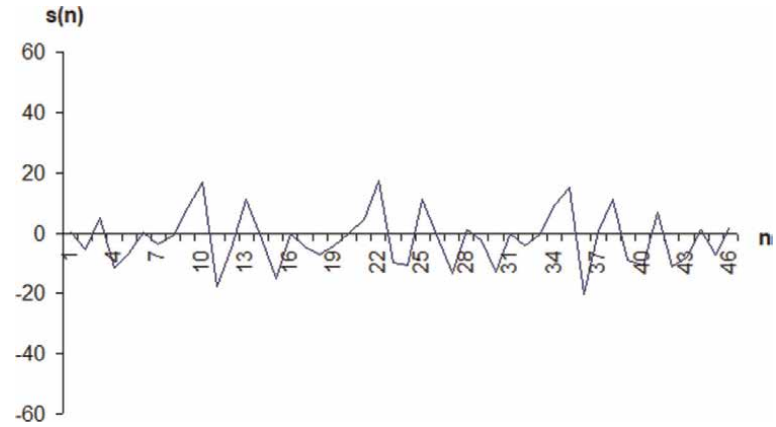

Figure 3.
The decimation signal.

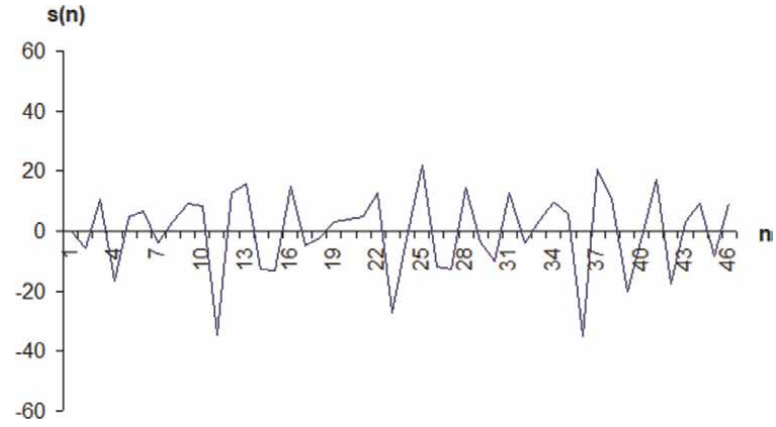

Figure 4.
A signal in the form of the weighed difference.

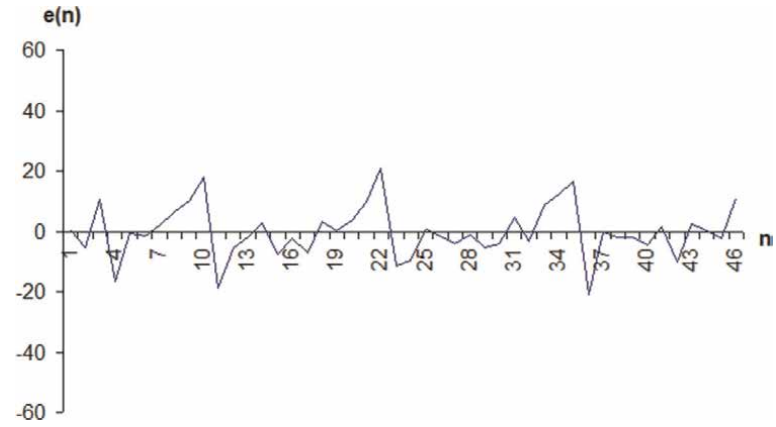

Figure 5.
Prediction error.

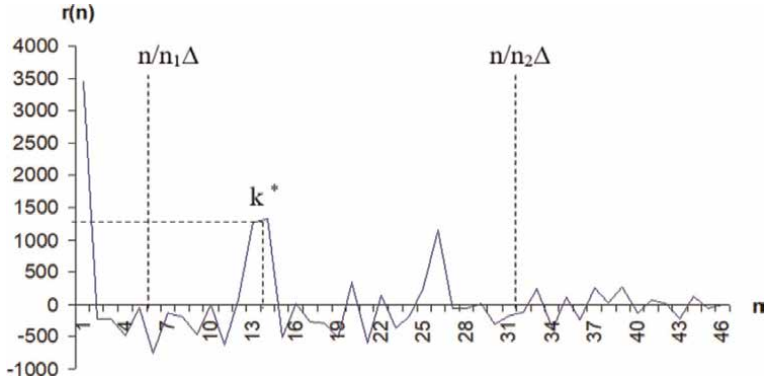

Figure 6.
Autocorrelated function of prediction error.

$$d_{\mu l} = \sum_{n=0}^{N-1} x(n)\ \ a_0{}^{-\mu/2}\overline{\psi(a_0{}^{-\mu}n - b_0 l)}\ \Delta t,\ \ l \in \overline{0, N-1},\ \ \Delta t = 1/f_d, \tag{19}$$

where μ—the decomposition level at which the smooth sinusoid is reached, N—signal length, Δt—quantization step.

For Morle's wavelet

$$\psi(\xi) = (2\pi)^{-1/2}\cos(k_0\xi)e^{-\xi^2/2},\ \ k_0 = 5,\ \ \xi = a_0{}^{-m}n - b_0 l. \tag{20}$$

As sequence $d_{\mu l}$ represents a smooth sinusoid, the use needs of autocorrelated function and function of the average value of a difference of signal amplitudes having considerable computing complexity disappears. Instead of calculation of these functions at the second stage in the sequence $d_{\mu l}$ two are defined in a row going a maximum and the difference between them in a form is calculated

$$\begin{aligned}&\Big(d_{\mu j-1} \le d_{\mu j} \ge d_{\mu,j+1} \wedge d_{\mu,m-1} < d_{\mu m} \ge d_{\mu,m+1} \wedge \\ &\wedge\big(d_{\mu,k-1} \ge d_{\mu k} < d_{\mu,k+1}\big) \wedge j < k < m\Big) \rightarrow k^* = m - j.\end{aligned} \tag{21}$$

The period of the fundamental tone is defined in a form

$$T_{\text{OT}} = \begin{cases} k^*, & k^* \in [n_2, n_2] \\ 0, & k^* \notin [n_2, n_2] \end{cases}, \tag{22}$$

where n_1—minimum length of the period of the fundamental tone, $n_1 = \inf T_{\text{OT}}$, n_2—maximum length of the period of the fundamental tone, $n_2 = \sup T_{\text{OT}}$.

Example 2

In **Figure 7** it is given a sound "A", and in **Figure 8**—a sound "A" on $\mu = 50$ decomposition level.

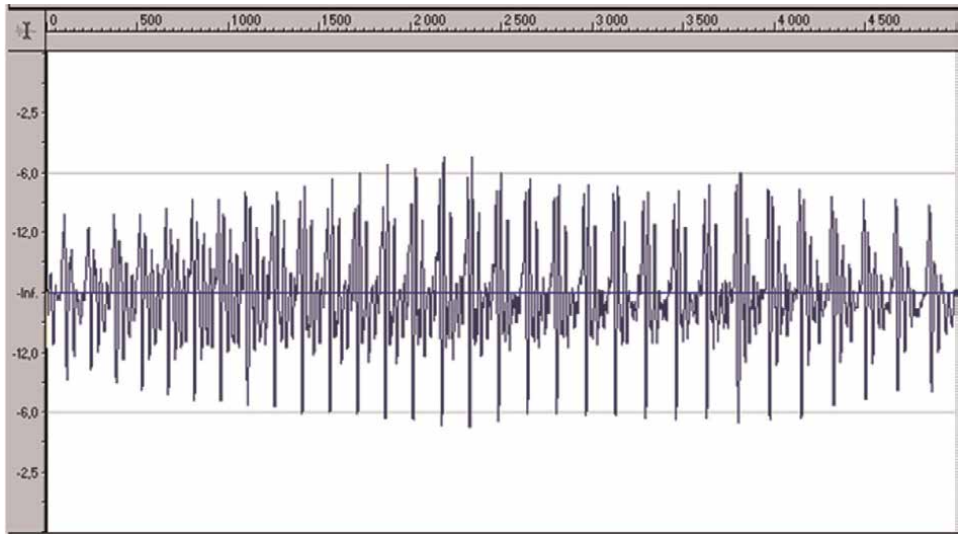

Figure 7.
Sound "A" for wavelet analysis.

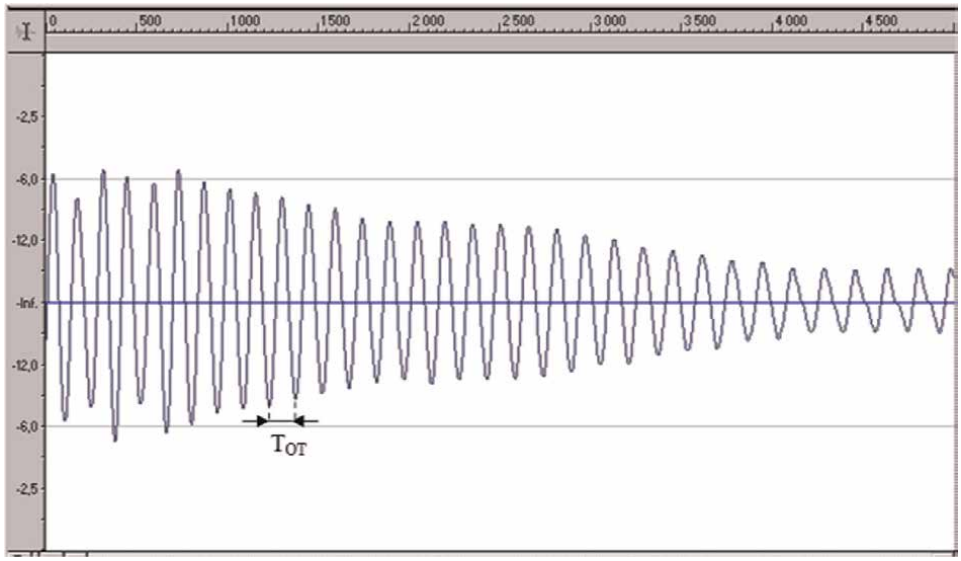

Figure 8.
A sound "A" at the 50th level of decomposition (frequency range is 51–250 Hz).

2.5 Method based on the cepstral analysis

This method carries out search of the maximum value in cepstrum [3].

1. For the chosen signal frame of length ΔN calculates a spectrum, using DFT

$$X(k) = \sum_{n=0}^{\Delta N-1} x(n) e^{-j(2\pi/\Delta N)nk}, \quad k \in \overline{0, \Delta N - 1}. \tag{23}$$

2. Cepstrum is calculated, using the inverse DFT

$$s(n) = \frac{1}{\Delta N} \sum_{k=0}^{\Delta N-1} \lg|X(k)|^2 e^{j(2\pi/N)nk}, \quad n \in \overline{0, \Delta N - 1}. \tag{24}$$

3. Is defined at what value n cepstrum $s(n)$ it is maximum that corresponds to extraction of the periods in a speech signal

$$n^* = \arg\max_n s(n), s^* = \max_n s(n), \quad n \in [n_1, n_2], \tag{25}$$

where n_1—minimum length of the period of the fundamental tone, $n_1 = \inf T_{\text{OT}}$, n_2—maximum length of the period of the fundamental tone, $n_2 = \sup T_{\text{OT}}$.

The period of the fundamental tone is defined in a form

$$T_{\text{OT}} = \begin{cases} n^*, & s^* \geq \gamma \\ 0, & s^* < \gamma \end{cases}, \tag{26}$$

where γ—the threshold value.

Example 3

As a signal the frame of a sound "A" length is chosen ΔN = 512 with a sampling frequency f_d = 22050 Hz, 8 bits, mono. In **Figure 9** it is given an initial signal, and in **Figure 10**—cepstrum of a signal.

2.6 HPS method

The harmonic product spectrum (HPS) method carries out search of the maximum value in the product of harmonicas of the decimated power spectrum [3].

1. For the chosen signal frame of length ΔN calculates a spectrum, using DFT

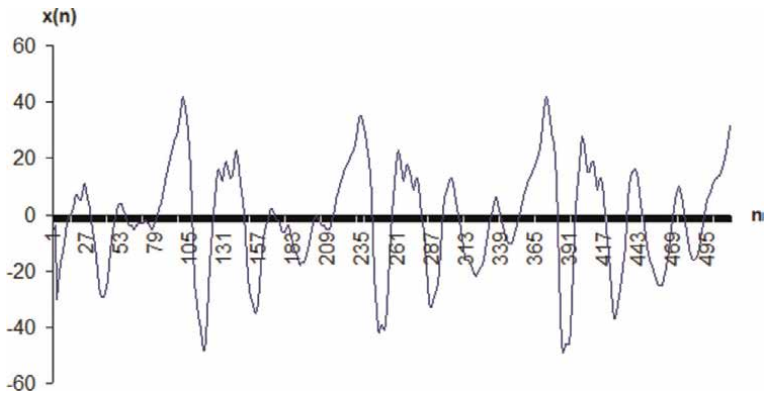

Figure 9.
Initial signal for cepstrum analysis.

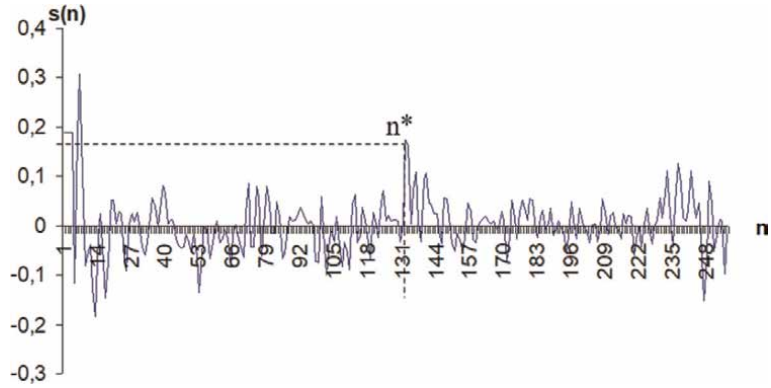

Figure 10.
Cepstrum of a sound "A".

$$X(k) = \sum_{n=0}^{\Delta N-1} x(n)e^{-j(2\pi/\Delta N)nk}, \quad k \in \overline{0, \Delta N - 1}. \tag{27}$$

2. The power spectrum of a signal is calculated

$$W(k) = |X(zk)|^2, \quad k \in \overline{0, \Delta N - 1}. \tag{28}$$

3. Z times a power spectrum of a signal is decimated, i.e. intermediate frequencies of a power spectrum of a signal are removed

$$W_z(k) = |X(zk)|^2, \quad k \in \overline{0, [\Delta N/z] - 1}, \quad z \in \overline{1, Z}, \tag{29}$$

where $[\cdot]$—integer part of number.

4. The product of harmonicas of the decimated power spectrum is calculated

$$P(k) = \prod_{z=1}^{Z} W_z(k), \quad k \in \overline{0, [\Delta N/Z] - 1}. \tag{30}$$

5. Is defined at what value k the product of harmonicas of the decimated power spectrum as much as possible that corresponds to extraction of the periods in a speech signal

$$k^* = \arg\max_k P(k), \quad k \in \overline{0, [\Delta N/Z] - 1}. \tag{31}$$

Frequency of the fundamental tone is determined in a form

$$F_{\text{OT}} = \begin{cases} k^*, & k^* \in [k_2, k_2] \\ 0, & k^* \notin [k_2, k_2] \end{cases}, \tag{32}$$

where k_1—minimum frequency of the fundamental tone, $k_1 = \inf F_{\text{OT}}$, k_2—maximum frequency of the fundamental tone, $k_2 = \sup F_{\text{OT}}$.

The SIFT, ACF, AMDF methods, based on the cepstral analysis depend on noise level.

The HPS methods, on a basis a wavelet analysis, are resistant to noise.

The SIFT methods, based on the cepstral analysis demand a threshold task.

The method on a basis a wavelet analysis demands the setting level of decomposition.

The HPS method demands a task of decimating quantity.

3. Calculation method of linear prediction parameters

The linear predictive coding method uses the amplifier and the digital filter (**Figure 11**).

Thus, the signal can be presented in the signal form at the input of the linear system with variables on time parameters excited by quasiperiodic impulses or random noise.

Transfer function of a linear system with variable parameters $H(z)$ is considered as the relation of an output signal spectrum $S(z)$ to input signal spectrum $U(z)$

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{A(z)}, \quad A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k}, \tag{33}$$

where $A(z)$—the inverse filter for the system $H(z)$, G—coefficient of gain, p—a prediction order (filter order).

The input signal $u(n)$ is presented by the pulse sequence and noise. The model has the following parameters: coefficient of gain G and coefficients of the digital filter $\{a_k\}$. All these parameters slowly change in time and can be estimated on frames.

This method as features linear prediction coefficients (LPC), reflection coefficients (RC), linear prediction cepstral coefficients (LPCC), log area ratio (LAR) coefficients are used [3].

1. Signal $s(m)$ breaks on L frames of the length ΔN. For n-th frame by means of LPF the balancing of the spectrum having steep descent in area of high frequencies is carried out

$$\breve{s}_n(m) = s_n(m+1) - \alpha s_n(m), \quad m \in \overline{0, \Delta N - 1}, \tag{34}$$

where α—filtration parameter, $0 < \alpha < 1$.

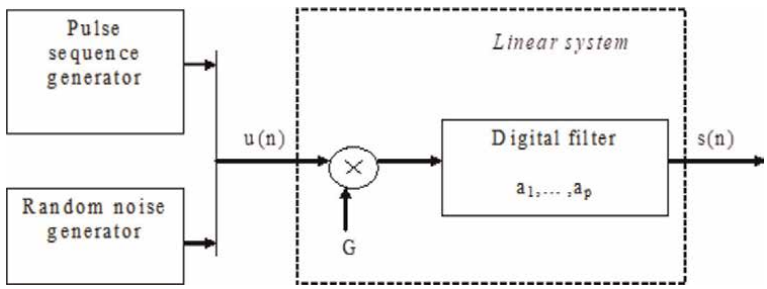

Figure 11.
The block diagram of the simplified model of signal formation.

2. For n-th frame the autocorrelated function is calculated $R_n(k)$

$$\widehat{s}_n(m) = \breve{s}_n(m)w(m), w(m) = 0.54 + 0.46\cos\frac{2\pi m}{\Delta N}, \tag{35}$$

$$R_n(k) = \sum_{m=0}^{\Delta N-1-k} \widehat{s}_n(m)\widehat{s}_n(m+k), \quad k \in \overline{0,p}, \tag{36}$$

where $w(m)$—Hamming's window, p—order of linear prediction, $ceil(f_d/1000) \le p \le 5 + ceil(f_d/1000)$, $ceil(f)$—function which rounds f to the next integer.

3. For n-th frame linear prediction coefficients (LPC) a_{nj} and reflection coefficients (RC) k_{ni} are calculated according to the procedure Darbin.

4. For n-th frame gain coefficient is calculated G_n.

$$G_n = \sqrt{E_n} = \sqrt{R_n(0) - \sum_{k=1}^{p} a_{nk}R_n(k)}. \tag{37}$$

5. For n-th frame linear prediction cepstral coefficients (LPCC) are calculated

$$LPCC_n(m) = \begin{cases} \ln G_n, & m = 0 \\ a_{nm}, & m = 1 \\ a_{nm} - \sum_{k=1}^{m-1}(k/m)LPCC_n(k)a_{n,m-k}, & 2 \le m \le p \end{cases}, \quad m \in \overline{0,p}. \tag{38}$$

6. For n-th frame log area ratio (LAR) coefficients are calculated

$$LAR_{nm} = \ln\left(\frac{1-k_{nm}}{1+k_{nm}}\right), \quad m \in \overline{1,p}. \tag{39}$$

4. Calculation method formant

For n-th of a frame the logarithmic power spectrum is calculated, using coefficient of gain and linear prediction coefficients (LPC) [3, 4]

$$10\lg W_n(k) = 10\lg\left|\frac{G_n}{A_n(z)}\right|^2 = \\ = 10\lg\frac{G_n^2}{\left(1 - \sum_{m=1}^{p} a_{nm}\cos\left(\frac{2\pi}{\Delta N}km\right)\right)^2 + \left(\sum_{m=1}^{p} a_{nm}\sin\left(\frac{2\pi}{\Delta N}km\right)\right)^2}. \tag{40}$$

At identification of the person or speech recognition for the analysis of vocalized sounds with a frequency range from 0 to 3 kHz are limited and the first 3 formant use F_1, F_2, F_3. At synthesis of the speech with a frequency range from 0 to 4–5 kHz are limited and use the first 5 formant F_1, F_2, F_3, F_4, F_5.

Example 4

In **Figure 12** the logarithmic power spectrum of the central frame of a sound "A" with different orders of prediction, at the same time length of a frame $N = 512$, sampling frequency is presented $f_d = 22050$ Hz.

Apparently from **Figure 12**, extraction a formant (maximum in a spectrum) perhaps already at $p = 30$.

Example 5

In **Figure 13** it is given a sound "A", and in **Figure 14**—its logarithmic power spectrum of LPC. In **Figure 15** it is given the central frame of a sound "Sh", and in **Figure 16**—its logarithmic power spectrum of LPC. At the same time length of a frame $N = 512$, sampling frequency $f_d = 22050$ Hz., 8 bits, mono, prediction order $p = 30$.

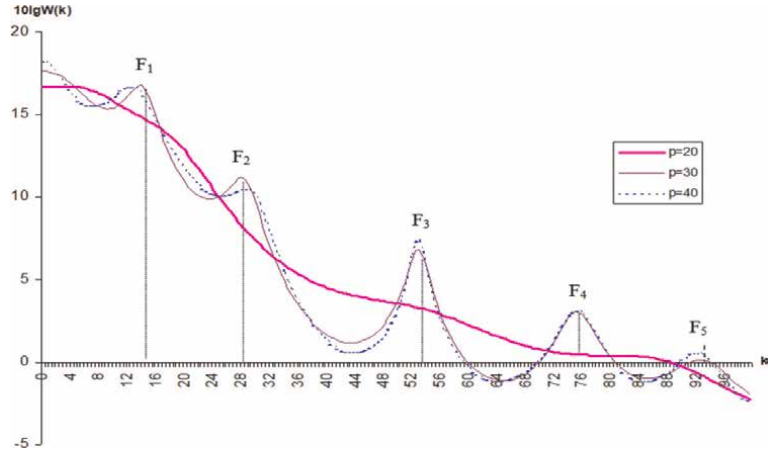

Figure 12.
The Logarithmic power spectrum of LPC of a sound "A" at different orders of prediction p.

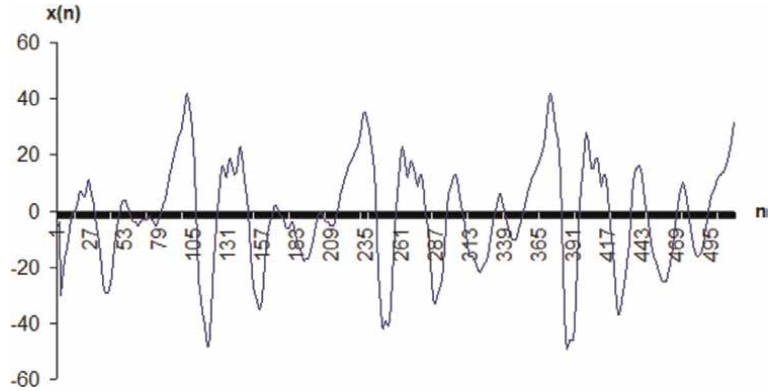

Figure 13.
Sound "A".

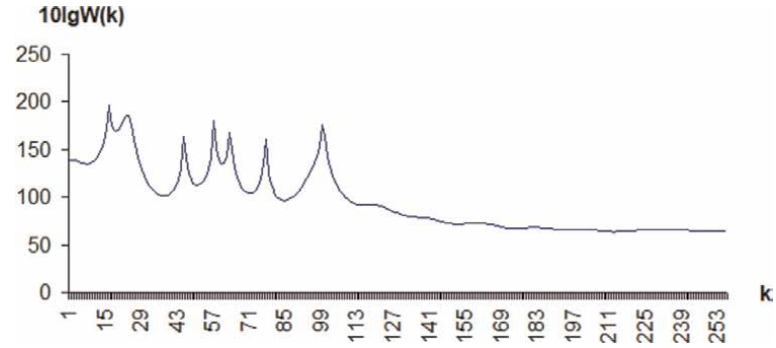

Figure 14.
Logarithmic power spectrum of LPC sound "A" at a prediction order p=30.

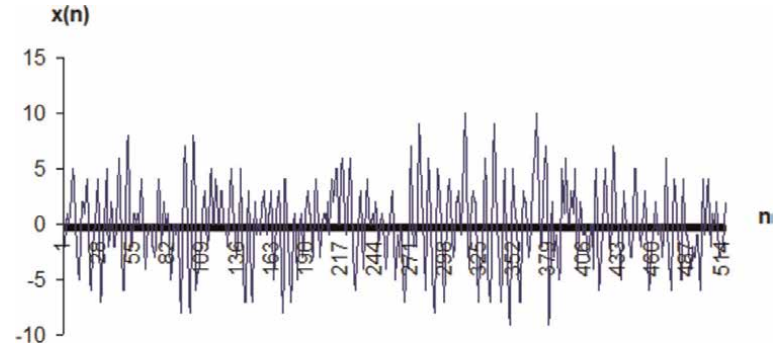

Figure 15.
Sound "Sh".

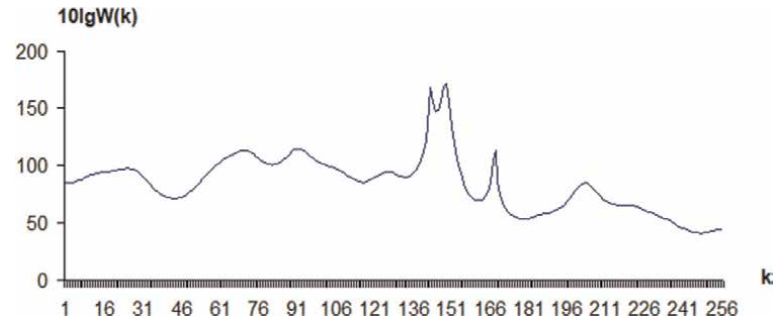

Figure 16.
Logarithmic power spectrum of LPC sound "Sh" at an order of prediction p=30.

5. Method of mel-frequency cepstral coefficients calculation

This method is based on homomorphic processing and uses as features mel-frequency cepstral coefficients (MFCC) [5, 6].

1. Signal $s(m)$ breaks on L frames of the length ΔN. For n-th frame by means of LPF the balancing of the spectrum having steep descent in area of high frequencies is carried out

$$\breve{s}_n(m) = s_n(m+1) - \alpha s_n(m), \quad m \in \overline{0, \Delta N - 1}, \tag{41}$$

where α—filtration parameter, $0 < \alpha < 1$.

2. For n-th frame the spectrum is calculated, using DFT

$$\widehat{s}_n(m) = \breve{s}_n(m) w(m), w(m) = 0.54 + 0.46 \cos \frac{2\pi m}{\Delta N}, \tag{42}$$

$$\widehat{S}_n(k) = \sum_{m=0}^{\Delta N-1} \widehat{s}_n(m) \; e^{-j(2\pi/\Delta N)km}, \quad k \in \overline{0, \Delta N - 1}, \tag{43}$$

where $w(m)$—Hamming's window.

3. For n-th frame on i-th mel-frequency band, the energy mel-frequency band is calculated, using frequency transformation and Bartlett's window

$$\widehat{E}_{nm} = \sum_{k=0}^{\Delta N/2-1} \left|\widehat{S}_n(k)\right|^2 w_m(k), m \in \overline{1, P}, w_m(k) = \begin{cases} 0, & k < f_{m-1} \vee k > f_{m+1} \\ \dfrac{k - f_{m-1}}{f_m - f_{m-1}}, & f_{m-1} \le k \le f_m \\ \dfrac{f_{m+1} - k}{f_{m+1} - f_m}, & f_m \le k \le f_{m+1} \end{cases}, \tag{44}$$

$$f_m = \frac{N}{f_d} B^{-1}\left(B\left(f^{\min}\right) + m \frac{B(f^{\max}) - B\left(f^{\min}\right)}{P+1}\right), \quad m \in \overline{0, P+1}, \tag{45}$$

$$B(f) = 1125 \ln(1 + f/700), B^{-1}(b) = 700(\exp(b/1125) - 1), \tag{46}$$

where $\widehat{E}_{im}$—energy of m-th mel-frequency band, $w_m(k)$—Bartlett's window for band m-th, $B(f)$—function which will transform frequency to Hz in frequency in mel, $B^{-1}(b)$—function which will transform frequency to mel in frequency in Hz, f_m—normalized frequency, $f^{\min}, f^{\max}$—minimum and maximum frequency in Hz (for example, $f^{\min} = 0, f^{\max} = f_d/2$), f_d—frequency of sampling of a speech signal in Hz, P—quantity of mel-frequency bands.

4. For n-th frame are calculated mel-frequency cepstral coefficients (MFCC), using the inverse discrete cosine transformation DCT-2

$$MFCC_n(m) = \sqrt{\frac{2}{P}} \sum_{k=0}^{P-1} \ln\left(\widehat{E}_{n,m+1}\right) \alpha(k) \cos\left(\frac{(2m+1)k\pi}{2P}\right) \; m \in \overline{0, \tilde{P} - 1},$$
$$\alpha(k) = \begin{cases} \sqrt{\frac{1}{2}}, & k = 0 \\ 1, & k > 0 \end{cases}, \tag{47}$$

where $\tilde{P}$—quantity mel-frequency cepstral coefficients, $1 \le \tilde{P} \le P$.

6. Method of bark-frequency cepstral coefficients calculation

This method is based on homomorphic processing and uses as features are used a bark-frequency cepstral coefficients (BFCC) [7, 8].

1. Signal $s(m)$ breaks on frames ΔN of length the L. For n-th frame the spectrum is calculated, using DFT

$$\widehat{s}_n(m) = s_n(m)w(m), w(m) = 0.54 + 0.46\cos\frac{2\pi m}{\Delta N}, \tag{48}$$

$$\widehat{S}_n(k) = \sum_{m=0}^{\Delta N-1} \widehat{s}_n(m)e^{-j(2\pi/\Delta N)km}, \quad k \in \overline{0, \Delta N-1}, \tag{49}$$

where $w(m)$—Hamming's window.

2. The quantity of bark-frequency bands is calculated

$$P = ceil\left(B\left(f_d/2\right)\right) + 1, B(f) = 6\text{asinh}(f/600), \tag{50}$$

where $ceil(f)$—function which rounds f to the next integer, f_d—frequency of sampling of a speech signal in Hz, $B(f)$—function which will transform frequency to Hz in frequency in a bark.

3. For n-th frame energy of bark-frequency bands is calculated

$$\widehat{E}_{nm} = \sum_{k=0}^{\Delta N/2-1} \left|\widehat{S}_n(k)\right|^2 w_m(k), \quad m \in \overline{0, P-1}, \tag{51}$$

$$b_m = m\frac{B\left(f_d/2\right)}{P-1}, \quad m \in \overline{0, P-1}, \tag{52}$$

$$\Delta b_{mk} = B\left(k\frac{f_d}{\Delta N}\right) - b_m, \quad m \in \overline{0, P-1}, \quad k \in \overline{0, \Delta N-1}, \tag{53}$$

$$w_m(k) = \begin{cases} 10^{(\Delta b_{mk}+0.5)}, & \Delta b_{mk} \le -0.5 \\ 1, & -0.5 < \Delta b_{mk} < 0.5 \\ 10^{-2.5(\Delta b_{mk}-0.5)}, & \Delta b_{mk} \ge 0.5 \end{cases}, \tag{54}$$

where $\widehat{E}_{im}$—energy of i-th a bark-frequency band, $w_m(k)$—trapezoidal window for band m-th.

4. For n-th frame the distortion of equal loudness for energy of bark-frequency bands is carried out

$$\breve{E}_{nm} = \left(v\left(B^{-1}(b_m)\right)\widehat{E}_{im}\right), \quad m \in \overline{0, P-1}, \tag{55}$$

$$B^{-1}(b) = 600\sinh\left(b/6\right), \tag{56}$$

$$v(f) = \begin{cases} \dfrac{\left(f^2 + 56.8 \cdot 10^6\right) f^4}{\left(f^2 + 6.3 \cdot 10^6\right)^2 \left(f^2 + 0.38 \cdot 10^9\right)}, & f_d < 5000 \\ \dfrac{\left(f^2 + 56.8 \cdot 10^6\right) f^4}{\left(f^2 + 6.3 \cdot 10^6\right)^2 \left(f^2 + 0.38 \cdot 10^9\right)\left(f^6 + 9.58 \cdot 10^{26}\right)}, & f_d \geq 5000 \end{cases}, \quad (57)$$

where $v(f)$—function for distortion of equal loudness (allows to approach human acoustical perception as the person has an unequal sensitivity of hearing at different frequencies), $B^{-1}(b)$—function which will transform frequency to a bark in frequency in Hz.

5. For n-th frame the law of intensity loudness is applied to energy of bark-frequency bands

$$\tilde{E}_{nm} = \left(\breve{E}_{nm}\right)^{0.33}, \quad m \in \overline{0, P-1}. \quad (58)$$

6. For n-th frame are calculated a bark-frequency cepstral coefficients (BFCC), using the inverse discrete cosine transformation DCT-2, and previously it is necessary to replace energy $\tilde{E}_{n0}$ and $\tilde{E}_{n,P-1}$ energy $\tilde{E}_{n1}$ and $\tilde{E}_{n,P-2}$ respectively

$$BFCC_n(m) = \sqrt{\frac{2}{P}} \sum_{k=0}^{P-2} \ln\left(\breve{E}_{n,m+1}\right) \alpha(k) \cos\left(\frac{(2m+1)k\pi}{2(P-1)}\right), \quad m \in \overline{0, \tilde{P}-1}, \quad (59)$$

$$\tilde{E}_{n0} = \tilde{E}_{n1}, \tilde{E}_{n,P-1} = \tilde{E}_{n,P-2}, \alpha(k) = \begin{cases} \sqrt{\frac{1}{2}}, & k = 0 \\ 1, & k > 0 \end{cases}, \quad (60)$$

where $\tilde{P}$—quantity a bark-frequency cepstral coefficients, $1 \leq \tilde{P} \leq P$.

7. Method of parameters of perceptual linear prediction calculation

In this method as features perceptual linear prediction coefficients (PLPC), perceptual reflection coefficients (PRC), perceptual linear prediction cepstral coefficients (PLPCC), perceptual log area ratio (PLAR) coefficients are used [9, 10].

1. Signal $s(m)$ breaks on frames ΔN of the length L. For n-th frame the spectrum is calculated, using DFT

$$\widehat{s}_n(m) = s_n(m) w(m), w(m) = 0.54 + 0.46 \cos\frac{2\pi m}{\Delta N}, \quad (61)$$

$$\widehat{S}_n(k) = \sum_{m=0}^{\Delta N-1} \widehat{s}_n(m) e^{-j(2\pi/\Delta N)km}, k \in \overline{0, \Delta N-1}, \quad (62)$$

where $w(m)$—Hamming's window.

2. The quantity of bark-frequency bands is calculated

$$P = ceil\left(B\left(f_d/2\right)\right) + 1, B(f) = 6\text{asinh}(f/600), \tag{63}$$

where $ceil(f)$—function which rounds f to the next integer, f_d—frequency of sampling of a speech signal in Hz, $B(f)$—function which will transform frequency to Hz in frequency in a bark.

3. For n-th frame energy of bark-frequency bands is calculated

$$\widehat{E}_{nm} = \sum_{k=0}^{\Delta N/2-1} \left|\widehat{S}_n(k)\right|^2 w_m(k), \quad m \in \overline{0, P-1}, \tag{64}$$

$$b_m = m\frac{B\left(f_d/2\right)}{P-1}, \quad m \in \overline{0, P-1}, \tag{65}$$

$$\Delta b_{mk} = B\left(k\frac{f_d}{\Delta N}\right) - b_m, \quad m \in \overline{0, P-1}, \quad k \in \overline{0, \Delta N-1}, \tag{66}$$

$$w_m(k) = \begin{cases} 10^{(\Delta b_{mk}+0.5)}, & \Delta b_{mk} \le -0.5 \\ 1, & -0.5 < \Delta b_{mk} < 0.5 \\ 10^{-2.5(\Delta b_{mk}-0.5)}, & \Delta b_{mk} \ge 0.5 \end{cases}, \tag{67}$$

where $\widehat{E}_{im}$—energy of i-th a bark-frequency band, $w_m(k)$—trapezoidal window for m-th band.

4. For n-th frame the distortion of equal loudness for energy of bark-frequency bands is carried out

$$\breve{E}_{nm} = \left(v\left(B^{-1}(b_m)\right)\widehat{E}_{im}\right), \quad m \in \overline{0, P-1}, \tag{68}$$

$$B^{-1}(b) = 600\sinh\left(b/6\right), \tag{69}$$

$$v(f) = \begin{cases} \dfrac{\left(f^2 + 56.8\cdot 10^6\right)f^4}{\left(f^2 + 6.3\cdot 10^6\right)^2\left(f^2 + 0.38\cdot 10^9\right)}, & f_d < 5000 \\ \dfrac{\left(f^2 + 56.8\cdot 10^6\right)f^4}{\left(f^2 + 6.3\cdot 10^6\right)^2\left(f^2 + 0.38\cdot 10^9\right)\left(f^6 + 9.58\cdot 10^{26}\right)}, & f_d \ge 5000 \end{cases}, \tag{70}$$

where $v(f)$—function for distortion of equal loudness (allows to approach human acoustical perception as the person has an unequal sensitivity of hearing at different frequencies), $B^{-1}(b)$—function which will transform frequency to a bark in frequency in Hz.

5. For n-th frame the law of intensity loudness is applied to energy of bark-frequency bands

$$\tilde{E}_{nm} = \left(\breve{E}_{nm}\right)^{0.33}, \quad m \in \overline{0, P-1}. \tag{71}$$

6. For n-th frame values of autocorrelated function are calculated, using the inverse DFT, previously it is necessary to replace energy $\tilde{E}_{n0}$ and $\tilde{E}_{n,P-1}$ energy $\tilde{E}_{n1}$ and $\tilde{E}_{n,P-2}$ respectively

$$R_n(k) = \text{Re}\left(\frac{1}{2P-2}\sum_{m=0}^{2P-3}\tilde{E}_{nm}e^{\,j(2\pi/M)km}\right), \quad k \in \overline{0, p}, \tag{72}$$

$$\tilde{E}_{n0} = \tilde{E}_{n1}, \tilde{E}_{n,P-1} = \tilde{E}_{n,P-2}, \tilde{E}_{n,2P-2-m} = \tilde{E}_{nm}, \quad m \in \overline{1, P-2}, \tag{73}$$

where p—order of linear prediction, $ceil\left(f_d/1000\right) \le p \le 5 + ceil\left(f_d/1000\right)$, $ceil(f)$—function which rounds f to the next integer.

7. For n-th frame perceptual linear prediction coefficients (PLPC) a_{nj} and perceptual reflection coefficients (PRC) k_{ni} are calculated according to the procedure Darbin.

8. For n-th frame gain coefficient is calculated G_n

$$G_n = \sqrt{E_n} = \sqrt{R_n(0) - \sum_{k=1}^{p} a_{nk}R_n(k)}. \tag{74}$$

9. For n-th of frame perceptual linear prediction cepstral coefficients (PLPCC) are calculated

$$PLPCC_n(m) = \begin{cases} \ln G_n, & m = 0 \\ a_{nm}, & m = 1 \\ a_{nm} - \sum\limits_{k=1}^{m-1}(k/m)PLPCC_n(k)a_{n,m-k}, & 2 \le m \le p \end{cases}, \quad m \in \overline{0, p}. \tag{75}$$

10. For n-th frame perceptual log area ratio (PLAR) is calculated

$$PLAR_{nm} = \ln\left(\frac{1-k_{nm}}{1+k_{nm}}\right), \quad m \in \overline{1, p}. \tag{76}$$

8. Method of parameters of reconsidered perceptual linear prediction calculation

In this method as features reconsidered perceptual linear prediction coefficients (RPLPC), reconsidered perceptual reflection coefficients (RPRC), the reconsidered perceptual linear prediction cepstral coefficients (RPLPCC), the reconsidered perceptual log area ratio (PLAR) coefficients are used [7, 8].

1. Signal $s(m)$ breaks on L frames of the length ΔN. For frame n-th by means of LPF the balancing of the spectrum having steep descent in area of high frequencies is carried out

$$\breve{s}_n(m) = s_n(m+1) - \alpha s_n(m), \quad m \in \overline{0, \Delta N - 1}, \tag{77}$$

where α—filtration parameter, $0 < \alpha < 1$.

2. For n-th frame the spectrum is calculated, using DFT

$$\widehat{s}_n(m) = \breve{s}_n(m) w(m), w(m) = 0.54 + 0.46 \cos \frac{2\pi m}{\Delta N}, \tag{78}$$

$$\widehat{S}_n(k) = \sum_{m=0}^{\Delta N - 1} \widehat{s}_n(m) e^{-j(2\pi/\Delta N)km}, \quad k \in \overline{0, \Delta N - 1}, \tag{79}$$

where $w(m)$—Hamming's window.

3. For n-th frame on i-th mel-frequency band, the energy mel-frequency band is calculated, using frequency transformation and Bartlett's window

$$\widehat{E}_{nm} = \sum_{k=0}^{\Delta N/2 - 1} \left| \widehat{S}_n(k) \right|^2 w_m(k), \quad m \in \overline{1, P}, \tag{80}$$

$$w_m(k) = \begin{cases} 0, & k < f_{m-1} \vee k > f_{m+1} \\ \dfrac{k - f_{m-1}}{f_m - f_{m-1}}, & f_{m-1} \le k \le f_m \\ \dfrac{f_{m+1} - k}{f_{m+1} - f_m}, & f_m \le k \le f_{m+1} \end{cases}, \tag{81}$$

$$f_m = \frac{N}{f_d} B^{-1}\left(B\left(f^{\min} \right) + m \frac{B(f^{\max}) - B\left(f^{\min} \right)}{P+1} \right), \quad m \in \overline{0, P+1}, \tag{82}$$

$$B(f) = 1125 \ln(1 + f/700), B^{-1}(b) = 700(\exp(b/1125) - 1), \tag{83}$$

where $\widehat{E}_{im}$—energy of m-th mel-frequency band, $w_m(k)$—Bartlett's window for band m-th, $B(f)$—function which will transform frequency to Hz in frequency in mel, $B^{-1}(b)$—function which will transform frequency to mel in frequency in Hz, f_m—normalized frequency, $f^{\min}, f^{\max}$—minimum and maximum frequency in Hz (for example, $f^{\min} = 0, f^{\max} = f_d/2$), f_d—frequency of sampling of a speech signal in Hz, P—quantity of mel-frequency bands.

4. For n-th frame values of autocorrelated function are calculated, using the inverse DFT

$$R_n(k) = \operatorname{Re}\left(\frac{1}{2P-2} \sum_{m=0}^{2P-3} \widehat{E}_{n,m-1} e^{j(2\pi/M)km} \right), \quad k \in \overline{0, p}, \tag{84}$$

$$\widehat{E}_{n,2P-m} = \widehat{E}_{nm}, \quad m \in \overline{2, P-1}, \tag{85}$$

where p—order of linear prediction, $ceil(f_d/1000) \le p \le 5 + ceil(f_d/1000)$, $ceil(f)$—function which rounds f to the next integer.

5. For n-th frame reconsidered perceptual linear prediction coefficients (RPLPC) a_{nj} and reconsidered perceptual reflection coefficients (RPRC) k_{ni} are calculated according to the procedure Darbin.

6. For n-th frame gain coefficient is calculated G_n

$$G_n = \sqrt{E_n} = \sqrt{R_n(0) - \sum_{k=1}^{p} a_{nk} R_n(k)}. \tag{86}$$

7. For n-th frame the reconsidered perceptual linear prediction cepstral coefficients (RPLPCC) are calculated

$$RPLPCC_n(m) = \begin{cases} \ln G_n, & m = 0 \\ a_{nm}, & m = 1 \\ a_{nm} - \sum_{k=1}^{m-1} (k/m) RPLPCC_n(k) a_{n,m-k}, & 2 \le m \le p \end{cases}, \quad m \in \overline{0, p}. \tag{87}$$

8. For n-th frame the reconsidered perceptual log area ratio (PLAR) are calculated

$$RPLAR_{nm} = \ln\left(\frac{1 - k_{nm}}{1 + k_{nm}}\right), \quad m \in \overline{1, p}. \tag{88}$$

9. The performance comparison of various features for person identification

For the speech signals containing vocal sounds the sampling frequency 8 kHz and the number of quantization levels 256 was established. Sample length of a vocal sound of the speech is equal to 256.

A numerical research results of LPC, RC, LPCC, LAR coefficients, MFCC, BFCC, PLPC, PRC, PLPCC, PLAR coefficients, RPLPC, RPRC, RPLPCC, RPLAR coefficients received by methods of coding and used for biometric identification of people from the TIMIT database on vocal sounds by means of the Gaussian mixed models (GMM) are presented in **Table 1**.

For coding methods for the analysis of a speech signal the filter order in case of linear prediction is equal 12, in case of perceptual linear prediction is equal 4, in case of the reconsidered perceptual linear prediction is equal 12, quantity mel-frequency bands equally 20, quantity a bark-frequency bands equally 17, the number of cepstral parameters based on subbands is equal to 13.

The result presented in **Table 1** shows that the largest probability of identification and the smallest number of coefficients are provided by coding of a vocal sound of the speech based on PRC.

Coefficient's type	Identification probability	Coefficients number
LPC	0.72	12
RC	0.96	12
LPCC	0.90	13
LAR coefficients	0.82	12
MFCC	0.97	13
BFCC	0.98	13
PLPC	0.74	4
PRC	0.98	4
PLPCC	0.92	5
PLAR coefficients	0.84	4
RPLPC	0.73	12
RPRC	0.97	12
RPLPCC	0.91	13
RPLAR coefficients	0.83	12

Table 1.
Numerical research results of the coefficients used for personality biometric identification.

10. Conclusion

The preliminary stage of the biometric identification is speech signal structuring and extracting features.

For calculation of the fundamental tone are considered and in number investigated the following methods of digital signal processing—ACF (autocorrelation function) method, AMDF (Average Magnitude. Difference Function) method, SIFT (Simplified Inverse Filter Transformation) method, method on a basis a wavelet analysis, method based on the cepstral analysis, HPS (Harmonic Product Spectrum) method. For speech signal extracting features are considered and in number investigated the following methods of digital signal processing—the digital bandpass filters bank; spectral analysis (Fourier's transformation, wavelet transformation); homomorphic processing; linear predictive coding. This methods make it possible to extract linear prediction coefficients (LPC), reflection coefficients (RC), linear prediction cepstral coefficients (LPCC), log area ratio (LAR) coefficients, mel-frequency cepstral coefficients (MFCC), bark-frequency cepstral coefficients (BFCC), perceptual linear prediction coefficients (PLPC), perceptual reflection coefficients (PRC), perceptual linear prediction cepstral coefficients (PLPCC), perceptual log area ratio (PLAR) coefficients, reconsidered perceptual linear prediction coefficients (RPLPC), reconsidered perceptual reflection coefficients (RPRC), reconsidered perceptual linear prediction cepstral coefficients (RPLPCC), reconsidered perceptual log area ratio (RPLAR) coefficients. Results of a numerical research of speech signal features extraction methods for voice signals people from the TIMIT (Texas Instruments and Massachusetts Institute of Technology) database were received. The features PRC proved to be the most effective.

DOI: http://dx.doi.org/10.5772/intechopen.104634

Author details

Eugene Fedorov[1*], Tetyana Utkina[1] and Tetiana Neskorodieva[2]

1 Cherkasy State Technological University, Cherkasy, Ukraine

2 Vasyl' Stus Donetsk National University, Vinnytsia, Ukraine

*Address all correspondence to: fedorovee75@ukr.net

References

[1] Oppenheim AV, Schafer RW. Discrete-Time Signal Processing. Upper Saddle River, NJ: Prentice Hall; 2010. p. 1108

[2] Mallat S. A Wavelet Tour of Signal Processing: Sparse Way. Bourlington, MA: Academic Press; 2008. p. 832. DOI: 10.1016/B978-0-12-374370-1.X0001-8

[3] Rabiner LR, Schafer RW. Theory and Applications of Digital Speech Processing. Upper Saddle River, NJ: Pearson Higher Education; 2011. p. 1042

[4] Markel JD, Gray AH. Linear Prediction of Speech. Berlin: Springer Verlag; 1976. p. 382

[5] Davis SB, Mermelstein P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustic, Speech and Signal Processing. 1980;**28**(4):357-366

[6] Ganchev T, Fakotakis N, Kokkinakis G. Comparative evaluation of various MFCC implementations on the speaker verification task. In: Proceedings of SPECOM 2005. Vol. 1. Patras, Greece; 2005. pp. 191-194

[7] Josef R, Pollak P. Modified feature extraction methods in robust speech recognition. In: Proceedings of the 17th IEEE Internations Conference Radioelektronika. Brno, Czech Republic: IEEE; 2007. pp. 1-4

[8] Kumar P, Biswas A, Mishra AN, Chandra M. Spoken language identification using hybrid feature extraction methods. Journal of Telecommunications. 2010;**1**(2):11-15

[9] Huang X, Acero A, Hon H-W. Spoken Language Processing: A Guide to Theory, Algorithm, and System Development. Upper Saddle River, NJ: Prentice Hall; 2001. p. 980

[10] Hermansky H. Perceptual linear predictive (PLP) analysis of speech. Journal of the Acoustical Society of America. 1990;**87**(4):1738-1752. DOI: 10.1121/1.399423

Chapter 4

Generalized Spectral-Temporal Features for Representing Speech Information

Stephen A. Zahorian, Xiaoyu Liu and Roozbeh Sadeghian

Abstract

Based on extensive prior studies of speech science focused on the spectral-temporal properties of human speech perception, as well as a wide range of spectral-temporal speech features already in use, and motivated by the time-frequency resolution properties of human hearing, this chapter proposes and evaluates one general class of spectral-temporal features. These features, intended primarily for use in Automatic Speech Recognition (ASR) front ends, allow different realizations of general time-frequency concepts to be easily implemented and tuned through a set of frequency and time-warping functions. The methods presented are flexible enough to allow evaluation of the relative importance of the spectral and temporal features and to explore the trade-off between time and frequency resolution. Extensive ASR experiments were conducted to evaluate various spectral-temporal properties using this unified framework.

Keywords: time-frequency, features, automatic speech recognition, basis vectors, front end

1. Introduction

As mentioned elsewhere [1], good features for automatic speech recognition include relevance, compactness, completeness, and robustness. That is, speech features should be closely related to speech production and understanding, should be small in number, represent as much speech information as possible, and should be little changed in the presence of noise or varying external conditions.

As these elements suggest, both productive and receptive aspects of speech science form the foundation for signal processing to extract speech features. Although receptive aspects of speech science are most directly relevant to speech features for ASR, speech production models for vocal tract configurations are also a plausible starting point for guiding speech feature extraction. In terms of speech production, ever since the classic Peterson and Barney vowel study [2], by far the most widely used acoustic features for characterizing vocal tract shape are formants. For speech signal

processing applications, formant information is generally obtained by first modeling the vocal tract using an all-pole system, such as in the Perceptual Linear Predictive (PLP) front end [3]. The motivating idea is that nearly any transfer function can be approximated by a high-order all-pole model. Due to lack of automatic methods to reliably estimate formants [4], and also because formants cannot discriminate between speech sounds for which the main differences are unrelated to formants (such as fricatives) [5, 6], formants are rarely used as features for ASR. For ASR the all-pole approximation to the vocal tract is more typically replaced with cepstral features [7], which encode the global spectral envelope shape without any emphasis given to spectral peaks.

There are many complex issues raised in the speech science literature about receptive aspects of human speech that could be potentially taken into account for extracting speech features for use in ASR. However, the only effects taken into account for the features presented in this chapter are the primary considerations of frequency and temporal resolution.

Auditory processing research related to the cochlea's frequency selectivity provides the fundamental theory for auditory filterbanks, which are often used as a signal processing step to compute features for ASR. Many canonical studies, such as [8–10], have pointed out that humans discern low frequency components in a complex sound with much higher resolution than is the case for high frequencies. Hence, in speech front ends, to mimic this property, the physical frequency range is mapped to a perceptual scale, typically using bandpass filtering with 25–60 overlapping bands, each corresponding to approximately equal length regions along the cochlear membrane. The bandwidths are designed to match the frequency resolution at each center frequency. Various perceptual scales have been developed, such as the Mel scale [9], Bark scale [10, 11], and Equivalent Rectangular Bandwidth (ERB) scale [12].

Commonly used filterbanks include triangular filters [13] based on the Mel scale, trapezoidal filters [3] based on the Bark scale, and gammatone filters [14, 15] based on the ERB scale. The output power of each filterbank channel is computed as a weighted sum of the magnitude-squared Short Time Fourier Transform (STFT), weighted by the channel frequency response, and then amplitude scaled to approximate perceptual loudness, which is linearly proportional to the neuron firing rate of the auditory nerves [16]. The amplitude-scaled outputs are usually combined with a cosine transform to form cepstral features such as the widely used MFCC features [13]. Another front end for computing speech features is PLP [3]. In PLP an equal-loudness compensation is also modeled to account for the non-equal amplitude sensitivity of human hearing at different frequencies [17]. Motivated by the importance of formants, linear prediction coefficients are computed from the Bark domain spectrum using Durbin's recursive method [18] and then converted to cepstral features.

Figure 1 depicts static feature extraction for the MFCC and PLP front ends. Note that the expression static features refers to features computed from a single very short segment of speech (on the order of 20 ms duration), called a frame. These features are computed for each frame with frames typically spaced apart by approximately 10 ms, thus also overlapped by 10 ms. This gap between adjacent frames is the frame spacing. Static features based on perceptual frequency scales do not do not explicitly encode spectral trajectories over time. In [19–22] approximations of time "derivatives" of the static features are computed and appended to static features to reduce ASR error rate considerably (empirically on the order of 20%). These time derivatives are called

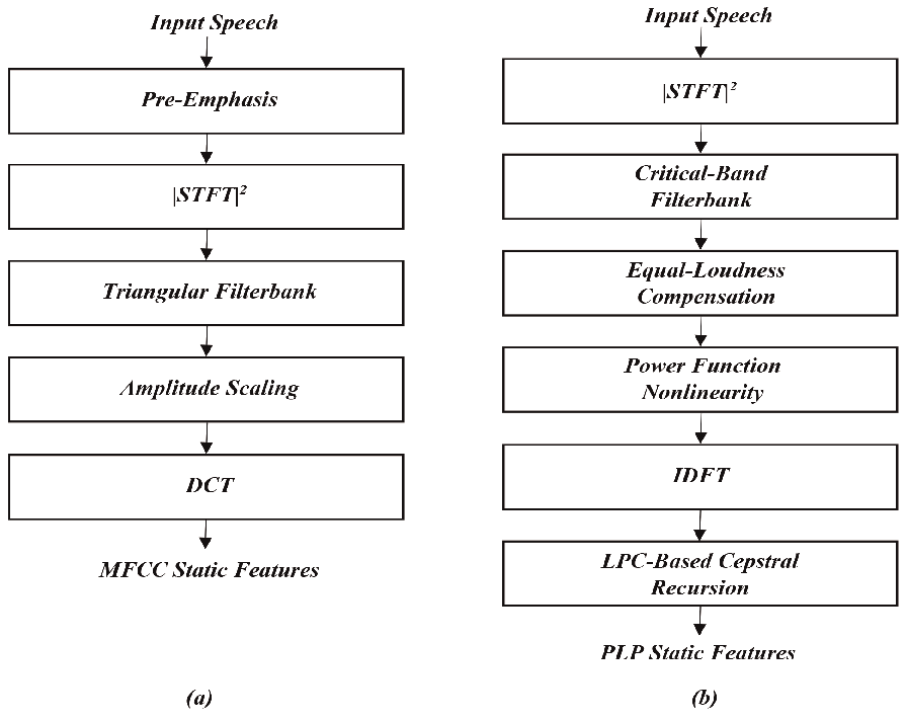

Figure 1.
Comparison of the MFCC (a) and PLP (b) structure.

dynamic features and are also often referred to as delta and acceleration (second order differential) terms. Mathematically, the delta terms are computed as:

$$\Delta_t = \frac{\sum_{\theta=1}^{\Theta} \theta(c_{t+\theta} - c_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2} \tag{1}$$

Where Δ_t is the differential at time t estimated from small adjacent groups of static features (cepstrums) $c_{t-\theta}$ to $c_{t+\theta}$ with $2\Theta + 1$ being the total number of surrounding frames. In the remainder of this chapter, groups of frames used to compute dynamic features from static features are referred to as blocks. More detailed discussion of frames and blocks, specifically related to the spectral-temporal features presented in this chapter, is given in Section 2.

Note that although the time derivatives are estimated from a short block of features, they essentially characterize the spectral trajectory at each single time instant, and thus are unable to account for the non-uniform time resolution of the human auditory system observed over a long duration of time. Spectral-temporal modulation features are much more effective than the delta method in addressing the issue of non-uniform time-frequency resolution and efficiently sampling the short-time spectrum. In 1994 Drullman et al. [23] found that the most important spectral trajectory information over time for speech perception is in the range of 1–16 Hz "modulation" frequencies. Guided by this finding, in order to exploit the information in the modulation frequencies, relatively long time blocks of each spectral band are analyzed. Over many years, various modulation features have been investigated.

Athineos et al. [24] used the dual of time-domain linear prediction to frequency-domain model the poles of the temporal envelope in each sub-band. Valente and Hermansky [25] developed an approach combining independent classifier outputs and modulation frequency channels. Gabor-filter-based approaches for extracting localized directional features also show promise [26, 27]. However, the large number of parameters, which allow Gabor filters to be aligned in many different directions, presents the added difficulty of determining these directions in an effective way for use in ASR.

Based on this prior extensive groundwork, this chapter presents a generalized spectral-temporal feature extraction front end for representing speech information. This feature set encompasses a wide range of time-frequency representation options focusing on two important properties of human hearing–frequency and time resolution. Rather than presenting one specific type of front end, a unified framework is presented such that various realizations of the general time-frequency concepts can easily be implemented and tuned. Based on a set of frequency-warping and time-warping functions, this front end is flexible enough to allow straightforward evaluation of the trade-off between frequency and time resolution at the acoustic feature level.

2. Method

The spectral-temporal features presented in this chapter are weighted sums of short-time spectral magnitudes, using overlapping frame-based processing. **Figure 2** illustrates the division of the short-time spectrum. The horizontal and vertical axes represent physical time (in seconds) and physical frequency (in Hz). A time-frequency representation (TFR) of the speech, denoted by $X(t,f)$, is obtained by computing the magnitude-squared STFT of each frame. In **Figure 2**, the dots in each

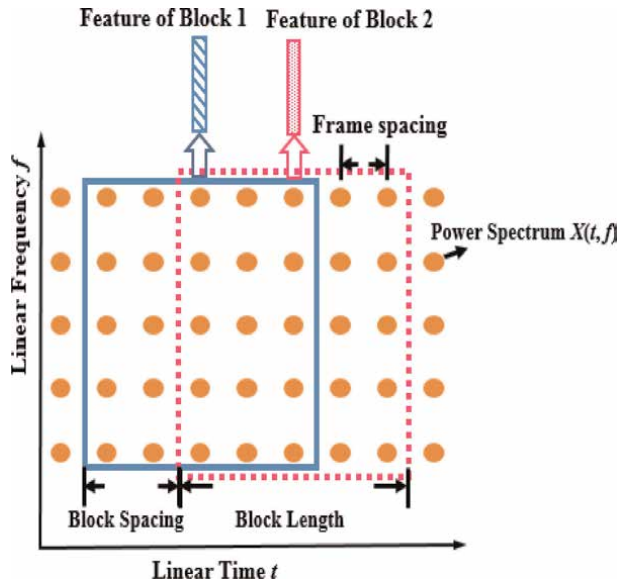

Figure 2.
A high level illustration of the proposed front end and definitions of related terminologies.

column represent the power spectrum of a frame, and the gap between adjacent columns denotes the frame spacing. Note that unlike the MFCC or PLP front ends, for which each feature vector is the concatenation of the spectral (static) feature and the spectral trajectory (dynamic feature) components, and the spectral trajectory is characterized by the time derivatives of the static terms at each sample instant on a frame-by-frame basis, in the method presented in this chapter, the front end computes a set of spectral-temporal features for a long block of spectral values centered at each sample instant, and one feature vector is extracted for each block. As will be seen in the derivations, this spectral-temporal feature vector for each block integrates both the spectral and temporal aspects of the speech signal within the block by a weighted sum of *X(t,f)* based on a set of two-dimensional spectral-temporal basis vectors. Thus, in the proposed front end, there are no individual static components in the final features since they are fused in the output features. Also, the use of long segments to compute features, using short highly overlapped frames, non-uniform time resolution can be incorporated in spectral trajectories.

Two basic concepts are also illustrated in **Figure 2**, which are used and referred to in the remainder of this chapter–block length and block spacing. Block length is defined as the time duration (physical time) of a block of short-time frames. Block length is measured in milliseconds and is equal to the frame spacing multiplied by the number of frames in the block. The spacing between two adjacent blocks is defined as block spacing, which is the product of the frame spacing and the number of frames that separate the two blocks. Since features are extracted on a block basis, the block spacing is also the feature spacing. At the beginning and ending of each speech utterance, zero padding is used to allow the first and last blocks to be centered at the first and last frames respectively. As opposed to MFCC or PLP processing, in which the feature spacing is identical to the frame spacing, in our work the feature spacing is typically considerably larger than the frame spacing. With these high level concepts, a detailed illustration of the feature extraction process is presented in the remainder of this section.

The time-frequency plane obtained by STFT has uniform frequency and time resolution determined by the analysis window shape and width [28]. This representation does not take into account the non-uniform perceptual frequency scale of the peripheral auditory system. For convenience and clarity of explanation, a framework is established with t' and f' as normalized perceptual time and frequency scales, whose desirable properties are next described in detail. Then a set of features, *Feat(i,j)* for the time block centered at time instant t, can be expressed as:

$$Feat(i,j) = \int_{t'=-\frac{1}{2}}^{\frac{1}{2}} \int_{f'=0}^{1} a\left(X'\left(t',f'\right)\right) \cdot BV_{i,j}\left(t',f'\right) df' dt'. \tag{2}$$

In Eq. (2) the feature computation is performed using perceptual scales, where $X'\left(t',f'\right)$ is the power spectrum of a time-frequency block in this domain for which the frequency f' is mapped to the range of {0, 1} by subtracting an offset and dividing by a scaling factor. Similarly, perceptual time t' is converted to the range of {−1/2, 1/2} with $t' = 0$ the center of the time block. The function $a(\cdot)$ nonlinearly maps the power spectrum to a perceptual-loudness scale, most often using a logarithmic scaling or a power-law nonlinearity [29]. Finally, the amplitude-scaled power spectrum is weighted by a set of two-dimensional basis vectors $BV_{i,j}$ in the perceptual domain $\left(t',f'\right)$. The number of features extracted from a time-frequency block depends on the number of basis vectors used.

It should be emphasized, that for clarity of explanation, integrals as well as continuous time and frequency variables are used in Eq. (2) in all of the following equations. In actual implementations, both time and frequency variables are discrete, as shown in **Figure 2**, and integrations are computed as sums. Also, although the feature extraction is effectively performed in the perceptual time-frequency domain (t', f'), the actual computations use the linear time-frequency plane. The mapping between linear and perceptual domains for time and frequency are established by nonlinear time and frequency-warping functions and incorporated by changes in underlying basis vectors as explained below.

In this work, a set of two-dimensional cosine basis vectors for $BV_{i,j}(t', f')$ is used to compactly encode the spectral envelope as well as the spectral trajectory. The theoretical work of Rao and Yip [30] gives reasons why the cosine transform is particularly appropriate for data compression and feature de-correlation, based on similarity to the data-driven Karhunen-Loeve Transform. For similar reasons, the MFCC features also use a one-dimensional cosine transform as a processing step. The popular JPEG standard for image compression also uses two-dimensional cosine transforms.

Continuing with the specifics of the method presented in this chapter, the 2-D cosine basis vectors operating in the perceptual space are defined as:

$$BV_{i,j}(t', f') = \cos(\pi i f') \cdot \cos(\pi j t'), \quad 0 \le i \le N-1, 0 \le j \le M-1. \tag{3}$$

Eq. (3) shows that each 2-D basis vector is the product of two individual basis vectors, one over frequency f', and one over time t'. The numbers of basis vectors over frequency and time are specified by N and M respectively. The total number of features for each block is given by N x M. As is discussed in detail in Section 3, a larger N or M provides a more detailed representation of the spectral envelope over frequency or the spectral trajectory over time respectively. Empirical data indicates a total of 75 features for each block (N = *15, M* = *5*) results in high ASR accuracy. Eqs. (4) through (9), and associated figures, show that the nonlinear mapping from f to f' and t to t', together with their differentials df' and dt', approximate the frequency and time resolution of human hearing. Next is shown how the nonlinear mappings are mathematically incorporated into the feature calculations. Frequency warping, specifies the relation between perceptual frequency f' and physical frequency f:

$$f' = g(f), 0 \le f \le 1 \tag{4}$$

The physical frequency range has also been normalized to {0,1}[1]. Thus, the df' term in Eq. (2) is equivalent to:

$$df' = \frac{dg}{df} df \tag{5}$$

[1] For convenience, the normalized frequency range {0,1} of f corresponds to the physical range {*0*, *Fs/2*} where *Fs/2* is the Nyquist frequency. The normalized perceptual frequency f' over {0,1} also represents the range of 0 to Fs/2. With minor changes, this normalized range can be reduced to a shorter frequency range of physical frequencies.

As per the discussion in Section I, one reasonable choice for the form of the frequency warping$g(f)$ is a Mel-shape warping defined as:

$$g(f) = C \cdot \log_{10}\left(1 + \frac{f}{k}\right) \tag{6}$$

where k is an adjustable warping factor between 0 and 1 that controls the degree of the warping, and the constant C is chosen to ensure that $f = 1$ is mapped to $f' = 1$. If $k = 0.0875$ and $C = 0.9137$, for the frequency range of 0 to 8000 Hz, this warping is the normalized version of the most widely used "standard" Mel warping proposed by O'Shaughnessy [31]. Another option, using Smith and Abel's work [32], is to use a bilinear warping to approximate the Bark scale:

The warping factor α ranges from 0 to 1. In **Figure 3**, five bilinear warpings, for various α values.are shown. Addiionaly, Mel warping using O′Shaughnessy's equation in [31] and Bark warping as per Wang et al. [33] are plotted in the figure. The figure clearly shows that bilinear warping can be adjusted to closely approximate both Mel and Bark warping. From Eq. (5), frequency resolution is continuosly varied, to match auditory properties, rather than using a quantized version with a filterbank, such as in the MFCC, PLP or gammatone front ends, [3, 13, 14]. In the filterbank methods, perceptually indistinguishable frequency components are modeled by the filter bandwidths. Thus, a filterbank is effectively a quantizer which separates the perceptual frequency scale into a finite number of equal intervals. In the proposed approach, the perceptual scale is continuous. The frequency selectivity is modeled by the derivative term $dg(f)/df$.

Next, the relation between perceptual timet' and linear time t is modeled with nonlinear (warping) function, h, but with a normalized range of t {−1/2, 1/2}:

$$t' = h(t,f);\ -\frac{1}{2} \le t \le \frac{1}{2}, 0 \le f \le 1. \tag{7}$$

Timet' can be considered a perceptual time scale that defines a "pseudo" time instant at which an acoustic event occuring at physical time t is perceived by the

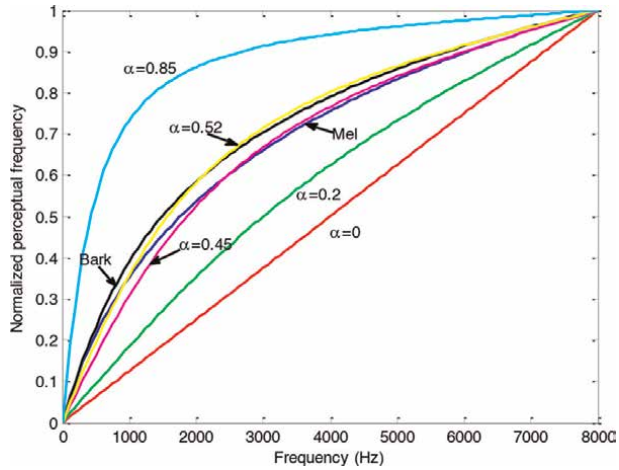

Figure 3.
Bilinear warping with different warping factors—Mel and Bark warping shown for comparison.

auditory system. Mathematically, perceptual time is given in terms of its derivative with respect to t:

$$dt' = \frac{dh(t,f)}{dt}\ \mathrm{dt} \tag{8}$$

This time resolution term indicates how far apart two events are perceived when separated by unit time on the physical scale. A large derivative implies that two acoustic events are clearly perceptually distinguishable whereas a small value corresponds to a time boundary between events that is not well resolved. When characterizing the temporal trajectory of acoustic events, it's reasonable to assume that perceptual time resolution should be higher near the center of the event than at far away times. That is, to identify the content of a segment with the help of its left and right segments, it is plausible that close segments are more relevant than far-away segments. Hence, temporal changes of the spectrum envelope should be more clearly resolved at the center of an event than far-away less helpful parts. Therefore, the shape for *dh/dt* was chosen to be approximately Gaussian. More specifically, *dh/dt* is a Kaiser window, with one parameter, β, the time-warping factor, that conveniently controls the "sharpness" of time warping.

Note that in Eqs. (8), (9) the sharpness of the time resolution term *dh/dt* could be frequency dependent as well. Specifically, the term *dh/dt* can be made more "peaky" at high frequencies than at low frequencies, controlled by different warping factor values in the Kaiser window[2], as illustrated in **Figure 4**. This allows an exploration of the trade-off between auditory frequency and time resolution. The psychoacoustic masking experiments [34] show that the very narrow auditory filter bandwidths at low frequencies produces high frequency resolution, but also prolongs the "ring" time at the onset and offset transients for short signals, and thus degrades the time resolution of the excitation patterns. This trade-off is also shown in [35] by

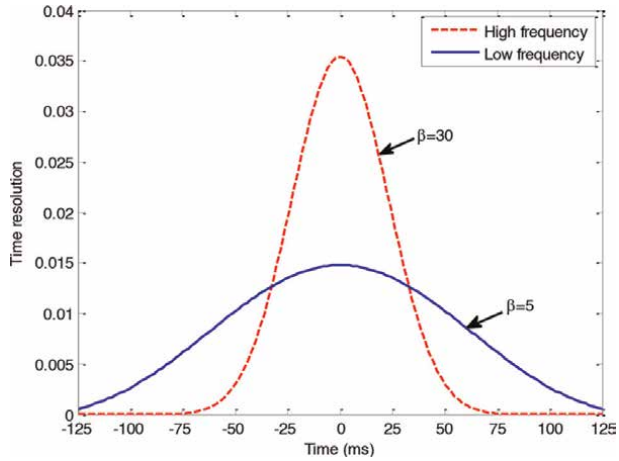

Figure 4.
Time resolution term dh/dt *for low and high frequencies using a Kaiser window: The time resolution is non-uniform over both time and frequency.*

[2] Note that although Eqs (8), (9) (and thereafter) explicitly show the frequency dependency in $h(t,f)$, in our implementation of $h(t,f)$ and its derivative, f is treated as a constant, and only t is the variable.

neurophysiological experiments and in [36] by the gap-in-noise detection experiments, which provide evidence that human subjects are able to detect shorter gaps in a narrow band of noise when the noise bands are centered at higher frequencies. Despite of this property of human hearing (high time resolution for high frequencies), it's not clear whether this effect can be exploited for improving ASR. Our work provides one way to investiage this effect in features used for ASR.

Although the principles and forms for frequency and time warping have been presented, the magnitude of the power spectrum on the perceptual scale is the same as for the.physical domain. To better represent perceptual magnitudes, the power spectrum should also be nonlinearly scaled. This nonlinear scaling is represented by the function a, typicaly logarithmic or power function with a low exponent such as 1/15. Eq. (2) can be rewritten in terms of t and f by substituting in Eqs. (3), (4), (5), (8), (9):

$$Feat(i,j) = \int_{t=-1/2}^{1/2}\int_{f=0}^{1} a(X(t,f)) \cdot \cos(\pi i g(f)) \frac{dg(f)}{df} \cdot \cos(\pi j h(t,f)) \frac{dh(t,f)}{dt} df dt \tag{9}$$

Eq. (10) can be written using modified basis vectors over frequency f as:

$$\varphi_i(f) = \cos(\pi i g(f)) \frac{dg(f)}{df}, \tag{10}$$
$$0 \le i \le N-1.$$

and modified frequency-dependent basis vectors over time t as:

$$\psi_j(t,f) = \cos(\pi j h(t,f)) \frac{dh(t,f)}{dt}, \tag{11}$$
$$0 \le j \le M-1.$$

Using the basis vectors from Eqs. (11), (12), Eq. (10) can be expressed as:

$$Feat(i,j) = \int_{t=-1/2}^{1/2}\int_{f=0}^{1} a(X(t,f)) \cdot \phi_{i,j}(t,f) df dt. \tag{12}$$

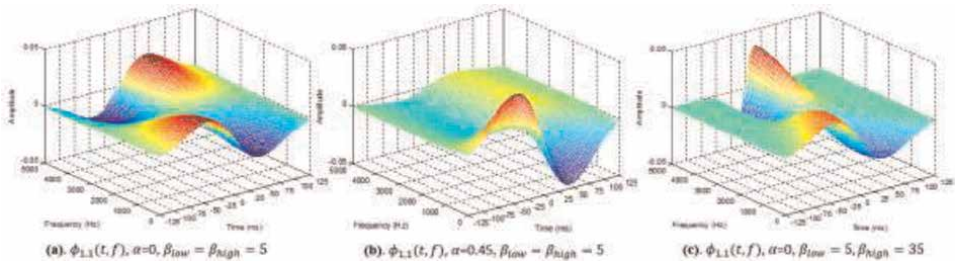

Figure 5.
Two-dimensional basis vector $\phi_{1,1}(t,f)$ with bilinear frequency warping g(f) *and a Kaiser window for dh(t,f)/dt.* α *is the frequency-warping coefficient as in Eq. (7), and* β_{low}, β_{high} *are the time-warping factors for low and high frequencies, respectively.*

where the two-dimensional basis vectors $\phi_{i,j}(t,f)$ are the product of the basis vectors given in Eqs. (11) and (12).

In **Figure 5**, the two-dimensional basis vector $\phi_{1,1}(t,f)$ is plotted with bilinear frequency warping *g(f)* and a Kaiser window for the *dh(t,f)/dt* term. Panels (a) and (b) are based on the same time-warping factor β = 5 for all frequencies, and only the frequency-warping factor α is varied. Compared with the linear frequency scale (α = 0) in panel (a), the basis vector becomes more sharply peaked at low frequencies in panel (b) as higher frequency resolution is incorporated through a larger warping factor α = 0.45. Panel (c) uses increasing time warping as frequency increases. The Kaiser window β value is linearly interpolated between β_{low} and β_{high}. The higher time resolution for high frequencies makes the basis vectors more concentrated near the center of the block.

Another option for the cosines used as the starting point for the two-dimensional basis vectors is to use a Gabor filterbank. As described in the work of [26, 27, 37], Gabor filtering is performed as a two-dimensional correlation between the Gabor filterbank and the perceptual time-frequency plane (t', f'). Each Gabor filter is defined using the product of a two-dimensional Gaussian envelope and a complex exponential function over a localized region in the time-frequency plane. Directionality is the most apparent difference between Gabor filter approach and the cosine expansion used in this chapter. Gabor filters can be adjusted toward any direction whereas the cosine transform only represents modulation of the spectrum along the vertical and horizontal axes. The deeper reason for this difference is that the Gabor approach and the method presented in this chapter are motivated by different considerations. The power spectrum directionality property of Gabor features stems from the response of neurons to combinations of spectral-temporal modulation frequencies in the spectral-temporal receptive field [38]. In contrast, the proposed framework is intended to model the trade-off between time and frequency resolution of the peripheral auditory system. However, it is possible to modify the proposed front end to incorporate the directionality of spectral-temporal patterns in a way similar to the Gabor filterbank. In prior work [39], this was achieved by rotating the 2-D cosine basis vectors by various angles.

3. Implementation

The 2-D integral in Eq. (10) can be implemented in a variety of ways, as discussed below. As mentioned previously, integrations are computed using sums and vector inner products between basis vectors and the sampled time-frequency plane.

3.1. DCTC/DCSC method

The first version of the implementation is based on frequency-independent time warping; i.e. the time warping *h(t,f)* is simplified to *h(t)* for all frequencies. In this case, integrating in any order (first over *f* and then over *t* or the reverse) is equivalent. Conventionally, frequency integration is performed first, which generates a set of intermediate static features[3] called Discrete Cosine Transform Coefficients (DCTCs):

[3] Note that the term "static features" refers only to the outputs of the DCTC step. As mentioned in the beginning of Section II, the final outputs are the spectral-temporal features, which are computed by another integration over the time sequence of these "static" features.

$$\mathrm{DCTC(i)} = \int_{f=0}^{1} \mathrm{a(X(t,f))} \cdot \varphi_i(\mathrm{f})\mathrm{df}, \tag{13}$$

where $\varphi_i(f)$ is the ith static basis vector as defined in Eq. (11). Then the trajectories of these DCTCs are encoded by integrations over time, yielding a set of features referred to as Discrete Cosine Series Coefficients (DCSCs):

$$DCSC(i,j) = \int_{t=-1/2}^{1/2} DCTC(i) \cdot \psi_j(t)dt, \tag{14}$$

where $\psi_j(t)$ is the jth basis vector over time, as defined in Eq. (12), but without dependence on f. These DCSC 2-D features, arranged as a 1-D feature vector, are the input to a recognizer. This implementation is depicted in **Figure 6(a)**. **Figure 7** is a plot of the first three DCTC and DCSC basis vectors, using a Mel-shape frequency warping and a Kaiser window with β = 5 for (derivative of) time warping. The zeroth order terms represent the form of the spectral/temporal resolution.

Unlike some other front ends, such as RASTA [40], TRAPS [41], as well as the Gabor method mentioned previously, for which modulation frequencies are a key concept, the proposed DCTC and DCSC method does not explicitly use this concept.

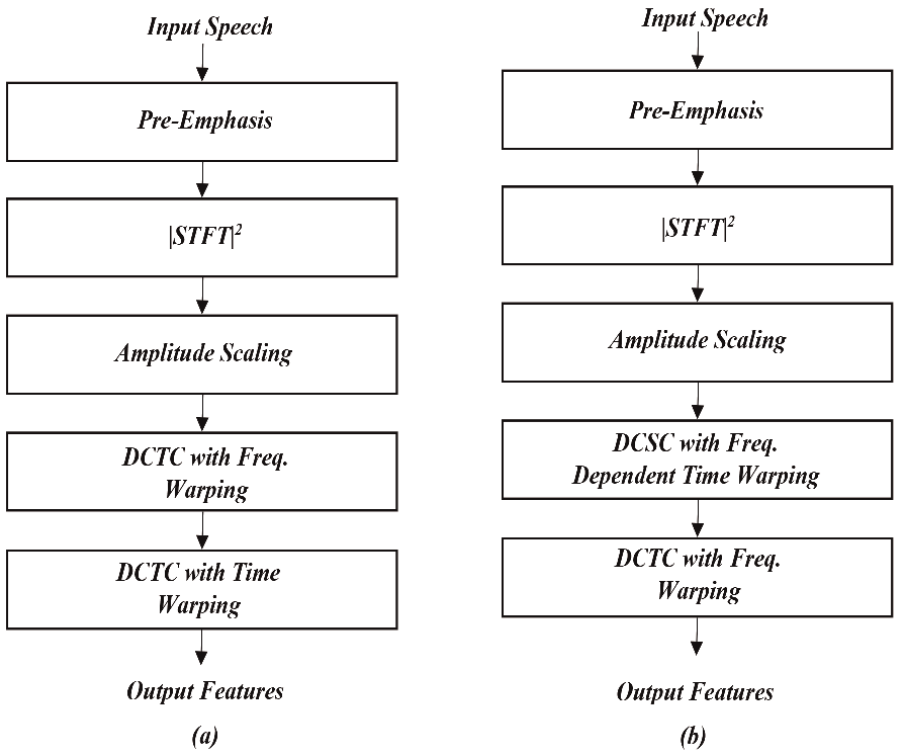

Figure 6.
Two implementations of the proposed front end: (a) the DCTC/DCSC implementation in which DCTCs are computed first followed by DCSCs. The time warping in the DCSC basis vectors is uniform for all frequencies. (b) the DCSC/DCTC implementation in which a set of DCSCs are obtained first followed by DCTCs. This implementation enables frequency-dependency in the DCSC basis vectors.

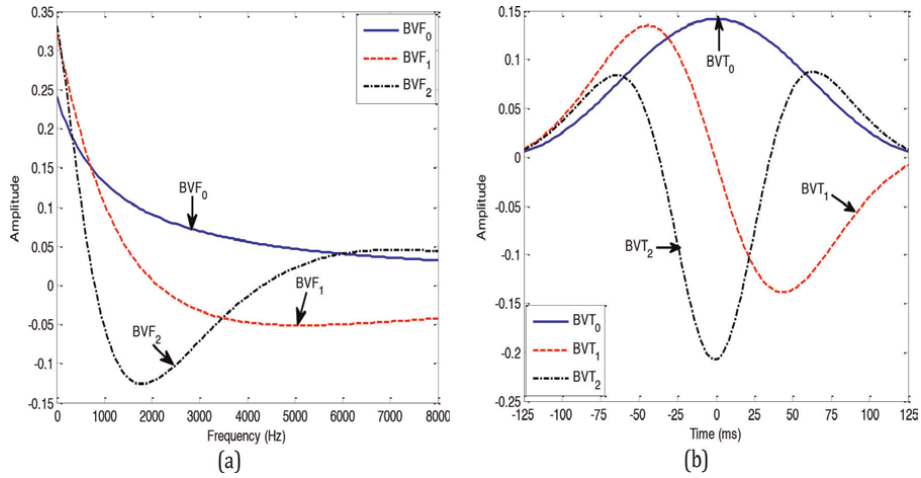

Figure 7.
The first three DCTC (a) and DCSC (b) basis vectors: A Mel-shape and a cumulative Kaiser window are used for frequency and time warping respectively.

The DCSC basis vectors act as non-causal FIR low pass temporal filters of spectral dynamics. Similarly, the DCTCs can also be viewed as low pass filtering of the power spectrum. Parameters used in the DCTC/DCSC implementation can be varied to examine the trade-offs between spectral and temporal resolution. The trade-off between spectral and temporal resolution considered here is different than the auditory time-frequency resolution built into the warping of the basis vectors as presented previously. Here, based on the filtering point of view, the parameters determine how much detail of the static spectrum and dynamic trajectory is preserved after the low pass filtering. The time-frequency resolution represented by the derivatives of the warping (which also cause a trade-off effect) is an intrinsic property of human hearing. As mentioned, the proposed DCTC/DCSC front end can be tuned to emphasize either side of the overall spectral or temporal resolution. For increased emphasis on the spectral information, a long frame length and a relatively large number of DCTCs should be used, with a relatively small number of DCSCs computed from a long block length. For increased emphasis on time resolution, a short frame length and frame spacing should be used with a large number of DCSCs computed from a short block length.

Figure 8 graphically illustrates this spectral-temporal trade-off. The top panel depicts the unprocessed spectrogram of a speech segment. Two spectrograms reconstructed from DCTC/DCSC terms are shown in the bottom panels[4]. The left one has high spectral resolution and low temporal resolution. It is rebuilt using 16 DCTCs, computed using 25 ms frames, a 10 ms frame spacing and 4 DCSCs with a block length of 50 frames (500 ms). The one in the right bottom panel has low spectral resolution but high temporal resolution. It is computed from 8 DCTCs, 5 ms frames spaced by 2 ms, and 6 DCSCs with a block length of 100 frames (200 ms). The low frequency components in both rebuilt

[4] Briefly, to rebuild the spectrum, the DCTCs and DCSCs are computed using orthonormal basis vectors, which can be obtained using Gram-Schmidt orthonormalization. Then the DCTCs of the center frame of a block are rebuilt first by multiplying the DCSCs by the transpose of the DCSC basis vector matrix and preserving only the center frame. Then the spectrum of this frame is rebuilt in a similar way by a matrix product using the transpose of the DCTC basis vector matrix.

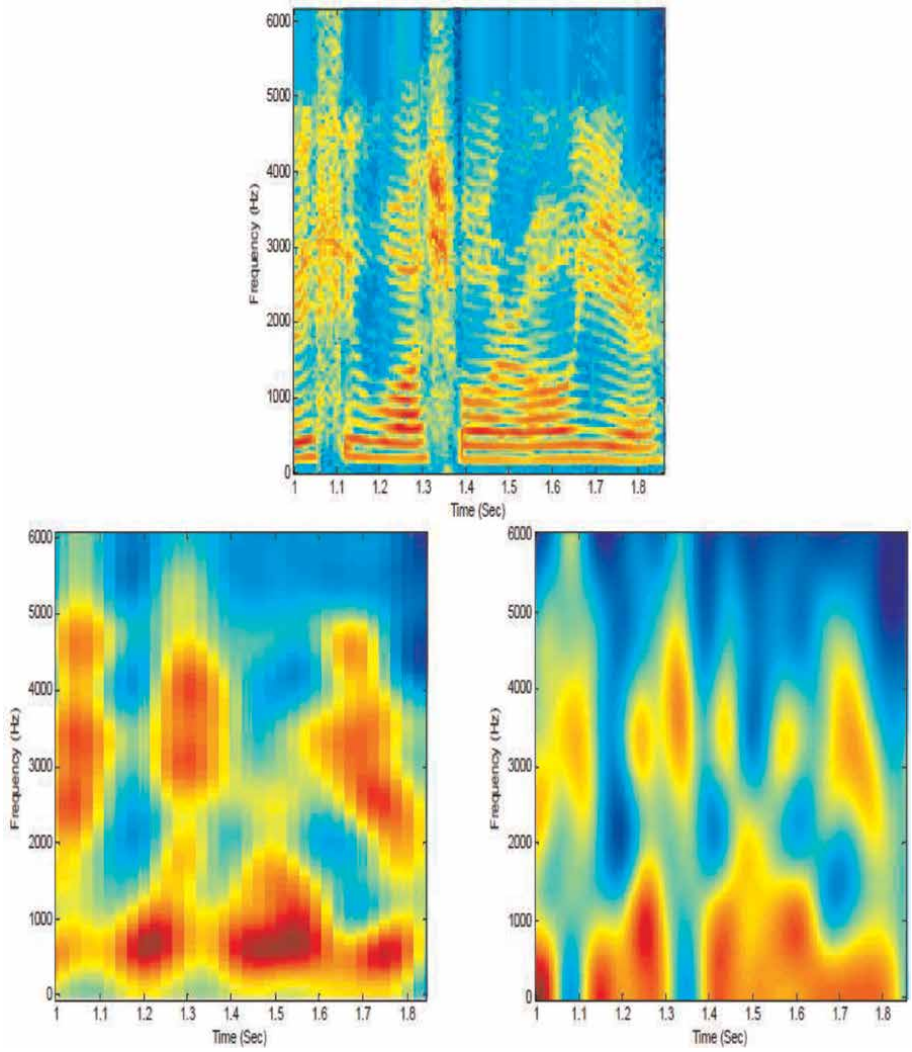

Figure 8.
Spectrogram of a speech segment (upper panel) and two rebuilt spectrograms: The bottom left one has high spectral resolution and low temporal resolution while the bottom right one has low spectral resolution but high temporal resolution.

spectrograms are represented with higher resolution than are the higher frequency components due to the Mel frequency warping. Comparing the two reconstructed spectrograms, the spectrogram in the left panel preserves more spectral details than does the spectrogram in the right panel. In contrast, the spectral dynamics are shown with more resolution in the right hand panel than are the spectral dynamics in the left pane.

3.2. DCSC/DCTC method

In the case of frequency-dependent time warping, the 2-D integration in Eq. (10) can be implemented by integrating over the time axis first followed by another integration over frequency. **Figure 6(b)** depicts the diagram of this configuration. In this case, Eq. (10) can be rearranged as:

$$Feat(i,j) = \left[\int_{f=0}^{1} \cos\left(\pi i g(f)\right) \frac{dg(f)}{df} \left[\int_{t=-1/2}^{1/2} a(X(t,f)) \cdot \cos\left(\pi j h(t,f)\right) \frac{dh(t,f)}{dt} dt\right] df\right] \tag{15}$$

The inner integral defines a set of frequency-dependent DCSCs,

$$DCSC(j,f) = \int_{t=-1/2}^{1/2} a(X(t,f)) \cdot \psi_j(t,f) dt, \tag{16}$$

where $\psi_j(t,f)$ is the jth DCSC basis vector for frequency f, as defined in Eq.(12). Then, the integral over frequency computes the DCTCs, which yields the final features

$$Feat(i,j) = DCTC(i,j) = \int_{f=0}^{1} DCSC(j,f) \cdot \varphi_i(f) df, \tag{17}$$

where $\varphi_i(f)$ is the ith DCTC basis vector as in Eq. (11).

3.3. Unified framework

As mentioned in Section 1, the DCTC/DCSC structure proposed in this chapter can be viewed as a unified framework which incorporates the filterbank implementation of the frequency warping as well as the conventional delta and acceleration dynamic features. To illustrate this unified viewpoint, a comparison of the "standard" MFCC front end and the DCTC/DCSC front end is presented in **Figure 9**.

In the filterbank-based front end, the frequency warping is performed by a group of auditory filters, and followed by a "regular" DCT transform with the term "regular" referring to sampled versions of half cosine basis vectors (in contrast to the basis vectors proposed in the previous sections). Specifically, the regular DCT transform is given by:

$$c(i) = \sqrt{\frac{2}{Q}} \sum_{j=1}^{Q} a(P(j)) \cos\left(\frac{\pi i}{Q}(j - 0.5)\right), \tag{18}$$

where $c(i)$ is the ith DCT coefficient, Q is the total number of filter channels, $P(j)$ is the output power of the jth channel, and $a(.)$ is the amplitude scaling function. The terms $\cos\left(\frac{\pi i}{Q}(j - 0.5)\right)$ are the unmodified cosine basis vectors.

In prior work [42], it was experimentally verified that the nonlinear amplitude scaling in the filterbank based front end can be moved to immediately before the filterbank without degrading ASR performance (i.e. swap the position of the filterbank block and the amplitude scaling block in **Figure 9(a)**). Then the filterbank weights can be combined with the unmodified cosine basis vectors by a simple matrix multiplication. Mathematically, suppose each row of the matrix $\boldsymbol{W}$ contains the magnitude response of a filterbank channel (i.e. if 26 channels are used with 128 FFT samples for each channel, $\boldsymbol{W}$ is a 26 by 128 matrix), and each row of the matrix $\boldsymbol{BVF}_{reg}$ contains the 12 unmodified cosine basis vectors, $\boldsymbol{BVF}_{reg}$ is a 12 by 26 matrix). A set of unified static basis vectors $\boldsymbol{BVF}_{uni}$, which incorporate the filterbank, can be formed by a matrix multiplication:

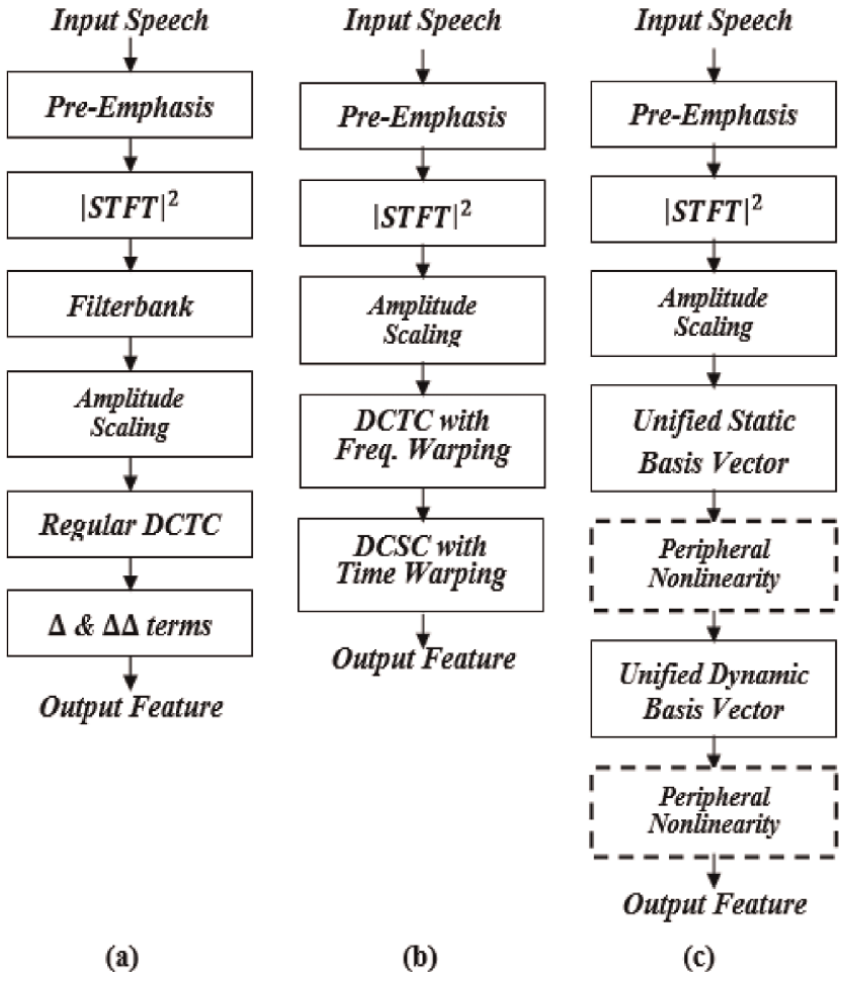

Figure 9.
Block diagrams of the filter bank front end (a), the DCTC/DCSC front end (b) and a unified framework (c) of (a) and (b) dashed blocks (– –) are optional.

$$\mathrm{BVF}_{\mathrm{uni}} = \mathrm{BVF}_{\mathrm{reg}} \cdot \mathrm{W} \tag{19}$$

In the proposed DCTC/DCSC case, $\boldsymbol{BVF}_{uni}$ is simply the matrix of the basis vectors $\varphi_i(f)$ defined in Eq. (11) with each row containing one such basis vector. Thus, with the unified static basis vectors, the static features in the filterbank front end and the DCTC/DCSC front end can be obtained using the same mathematical framework. The only difference lies in how their basis vectors are computed. Specifically, if the matrix $\boldsymbol{X}$ represents the power spectrum of a block of frames[5] for which each column is the magnitude squared STFT for a frame, the static features of this block for both the

[5] For consistency with the block processing in the computation of the dynamic features, the static feature computation also uses block notation here. When implemented, the static features are computed for the entire utterance once, and only the final features are computed block by block. That is, in the static feature step, $\boldsymbol{X}$ represents the spectrum of the entire utterance, and in dynamic feature step in Eq. (22), $\boldsymbol{X}$ denotes a block of frames.

filterbank front end and the DCTC/DCSC front end can be computed in a unified way as $\boldsymbol{BVF_{uni}} \cdot a(\boldsymbol{X})$ where $a(\boldsymbol{X})$ represents the amplitude scaling.

Similarly, the delta and higher order dynamics in the standard MFCC front end can also be computed by a summation of the static features over time, weighted by a set of dynamic basis vectors. From Eq. (1), to compute any nth order differential term, its basis vector with respect to the previous lower order terms (neglecting the constant denominator) is given by $\boldsymbol{bv}_n = [-\Theta_n, -\Theta_n + 1, \ldots, 0, 1, \ldots \Theta_n]$ where Θ_n is the window length in Eq. (1). Considering $\boldsymbol{bv}_n$ as a discrete signal with each element representing both the index and the amplitude (i.e. [−3,-2,-1,0,1,2,3] gives a signal whose magnitude is −3 at index −3, and − 2 at index −2, etc.), then the nth order delta basis vector $\boldsymbol{bvT_n}$ can be computed as

$$\mathbf{bvT_n} = \mathbf{bv_1} \circledast \mathbf{bv_2} \ldots \circledast \mathbf{bv_n}. \tag{20}$$

where ⊛ is the convolution operator. Thus, a set of unified dynamic basis vectors $\boldsymbol{BVT_{uni}}$ can be defined. In the case of the delta features, each row of $\boldsymbol{BVT_{uni}}$ stores one dynamic basis vector of the form in Eq. (21) whereas in the proposed DCTC/DCSC front end, each row of $\boldsymbol{BVT_{uni}}$ stores one DCSC basis vector as defined in Eq. (12). Hence, again the final output features $\boldsymbol{F}$ for both the MFCC and the DCTC/DCSC methods can be written in a unified way:

$$\mathbf{F} = \mathbf{BVT_{uni}} \cdot [\mathbf{BVF_{uni}} \cdot \mathrm{a}(\mathbf{X})]^{\mathrm{T}} \tag{21}$$

Figure 9(c) is a block diagram of this unified framework. This diagram depicts the essence of the proposed speech features as well as similar features such as MFCCs. They are essentially a series of linear transformations of the spectrum scaled by an auditory nonlinearity with optional peripheral nonlinearities in between (dashed blocks in the diagram), such as the sigmoid-shaped functions given in [43, 44]. These nonlinearities generally improve the noise robustness of front ends. In this work, the linear transformations are represented by unified basis vectors. Filterbank-based features (such as MFCC or PLP) exert their impact on features by shaping the basis vectors implicitly. The unified basis vectors presented here determine the properties of a front end. Thus we have a common yardstick with which to analyze and compare front ends based on the properties of the unified basis vectors.

A basic comparison can be made between filterbank-based frontends such as the widely-used MFCCs and the proposed DCTC/DCSC front end by comparing their unified basis vectors. Although the MFCC front end and the DCTC/DCSC front end are derived differently, the unified framework shows the two approaches are the same, except that the basis vectors are different.

Figure 10 is a plot of the first three unified static basis vectors underlying MFCC features (based on 26 Mel filters) and three unified temporal basis vectors used to compute the zeroth order, delta and acceleration terms. The unified basis vectors over frequency are not as "smooth" as the ones proposed here which are based on the continuous Mel-shape warping $g(f)$, as shown in **Figure 7(a)**. The "jagged" basis vectors 'plotted in **Figure 10(a)** result from the quantization effect caused by the coarse sampling of the frequency axis by the filter bank. The unified temporal basis vectors, implicit in most current methods, estimate derivatives very approximately using a small number of samples. A comparison of the temporal basis vectors (see **Figures 7(b)** and **10(b)**) graphically illustrate that the standard delta/acceleration method uses only a few central terms in each block whereas in the proposed method,

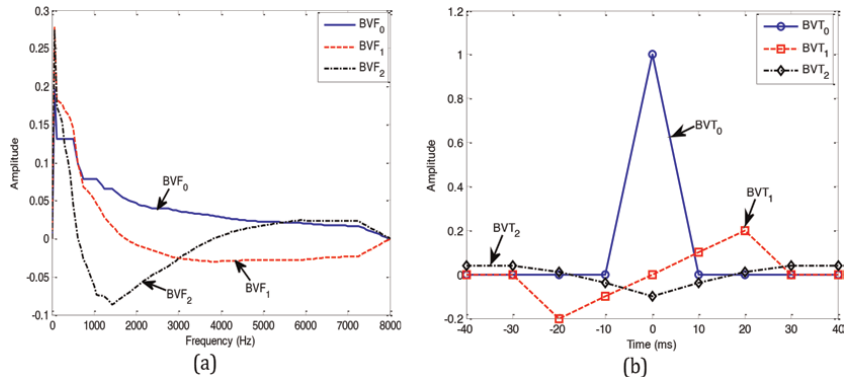

Figure 10.
The first three unified static basis vectors resulting from 26 Mel filters (a) and the first three unified dynamic basis vectors of the delta method (b).

the incorporation of non-uniform time resolution result in long "smooth" basis vectors emphasizing the center of the block but extending to the ends of the block. A comparison of both panels of **Figure 7** with both panels of **Figure 10** clearly illustrate the more continuous nature of the temporal basis vectors for the proposed method versus the implicit basis vectors corresponding to delta and acceleration terms. This suggests that the proposed DCSC basis vectors may represent spectral dynamics with more accuracy and resolution than is the case for delta/acceleration method.

4. Experimental evaluation

4.1 Experimental configuration

A comprehensive suite of ASR tests for various conditions and parameter settings was performed to evaluate the effectiveness of the spectral-temporal DCTC/DCSC features and to investigate trade-offs in time and frequency resolution as that affects ASR performance. All experiments reported in this chapter are for phone recognition, with monophone models using the HTK 3.4 HMM/GMM recognizer [45]. Except for one set of evaluation experiments described below, all experiments use the TIMIT database [46]. As is typically done with this database, 3696 utterances (462 speakers, eight sentences/speaker, approximately 236 minutes) with SA sentences removed were used for training. The TIMIT database document [46] suggests using 1344 utterances (168 speakers, eight sentences/speaker, approximately 86 minutes) for testing. However, since various parameters in the proposed front end needed to be tuned both for performance optimization and for exploring the effects on the time-frequency properties, a development set (DEV set) was needed. Thus, 672 utterances from the original test set were randomly chosen for this purpose, and the remaining 672 utterances were used as the evaluation set (EVAL set). Also, as recommended in [47], the original set of 61 labeled phones was collapsed to 48 phones to create 48 phone models with a further reduction to 39 phone categories for scoring. Some similar phones were merged to create the 39 categories: For convenient reference, the reduction from 61 to 48 phones and further from 48 to 39 phones (shaded) is presented in **Table 1**, and a frequency count of the 39 phones for the training and the original test sets is shown in **Figure 11**. All HMM acoustic models had three emitting

TIMIT Phone	Reduced Phone	TIMIT Phone	Reduced Phone	TIMIT Phone	Reduced Phone
oy	oy	v	v	er axr	er
uh	uh	b	b	iy	iy
aw	aw	f	f	r	r
th	th	w	w	el	el
ch	ch	ey	ey	l	l
y	y	ae	ae	ah	ah
jh	jh	dh	dh	ax-h ax	ax
g	g	d	d	s	s
ng eng	ng	p	p	en	en
sh	sh	eh	eh	n nx	n
zh	zh	m em	m	ih	ih
ow	ow	z	z	ix	ix
hh vv	hh	k	k	#h pau	sil
dx	dx	t	t	pcl tcl kcl qcl	cl
ay	ay	ao	ao	bcl dcl gcl	vcl
uw ux	uw	aa	aa	epi	epi

Table 1.
61 TIMIT phones, reduced to 48 for training, and 39 categories (shaded) for testing.

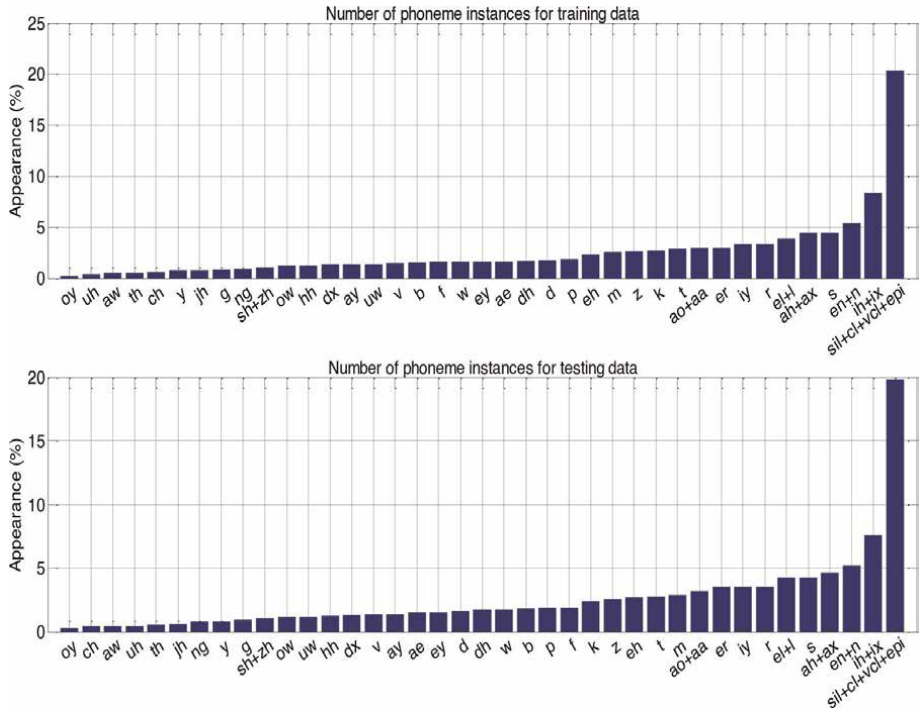

Figure 11.
A frequency count of the 39 TIMIT phone categories.

hidden states. A bigram language model was used based on phone bigram frequencies in the training set.

Since invariability is also crucial for "good" features, which means that the optimal front end parameters experimentally tuned from a DEV set should work approximately equally well on different independent evaluation sets without the need of re-tuning the parameters, an independent phone recognition task was also conducted using the Chinese Mandarin 863 Annotated 4 Regional Accent Speech Corpus (RASC863) [48]. The phonetically transcribed portion of this database was used for this work, which includes 20 speakers, each uttering 110 phonetically balanced sentences. Due to the much smaller number of speakers than for TIMIT, approximately 70% of the total set of 2200 utterances from all of the 20 speakers were used for training (1540 sentences, approximately 77 sentences/speaker and 224 minutes), and the remaining 30% were used for evaluation (660 sentences, 33 sentences/speaker, 96 minutes). Fifty-nine Chinese base phones (without considering tone information) were trained and evaluated on the evaluation set against the baseline directly using the optimal parameters obtained from the TIMIT experiments.

Processing begins with a complex pole pair IIR pre-emphasis filter:

$$y[n] = x[n] - 0.95x[n-1] + 0.494y[n-1] - 0.64y[n-2] \tag{22}$$

This second order filter has a peak near 3200 Hz and is a reasonably good match to the inverse of the equal-loudness contour for human hearing. In our previous work [49], it was found that this filter results in slightly higher ASR accuracy than is obtained with the more typically-used first-order one zero pre-emphasis. All speech passages were then divided into overlapping windowed frames (Kaiser window with β of 6, similar to a Hamming window). A 512 point FFT of each frame was computed, and log magnitudes computed, for a frequency range of 100 Hz to 7000 Hz. For each frame, log magnitudes were "floor" clipped at 40 dB below the largest spectral magnitude in each frame. In previous work [50], this simple floor was found to improve ASR accuracy by a small amount, especially for noisy speech. In summary, each sentence was converted to a matrix of spectral values which were then further processed by the DCTC/DCSC methods proposed in this chapter.

4.2 TIMIT DEV set parameter optimization

4.2.1 Experiment set 1: DCTC features only (static features)

For the experiments reported in this section, DCTC features only were computed. The goal was to experimentally evaluate how frame length, frame space, number of DCTCs, and type and degree of frequency warping affects ASR accuracy. Not all combinations of parameter values are presented in the results due to the very large number of combinations. Rather, most of the parameter values were fixed at what appeared to be the best values based on pilot experiments, and then a subset of parameter values was varied and performance evaluated.

Experiment A1— Spectral resolution issues for DCTCs: The goal was to examine spectral resolution effects on ASR performance as determined by frame length and number of DCTCs. The frame space was fixed at 8 ms. Mel frequency warping (bilinear warping with a coefficient of .45) and 16 mixture GMM/HMMs were used. The spectrum of each frame was represented with 9 to 26 DCTCs. Frame length ranged from 5 ms to 40 ms. ASR accuracy ranges from approximately 49–57% in these

tests. **Figure 12** depicts ASR accuracy using 21, 23, and 25 DCTCs as a function of frame length. It also contains the static MFCC baseline results using 26 filters, 21 DCTCs, again with the frame space fixed at 8 ms. The absolute best accuracy (57.3%) was obtained with 20 ms frames and 25 DCTCs. However, the increase in performance for more than 19 DCTCs is minimal, typically less than 0.5%. Frame lengths of 15 ms to 30 ms result in fairly similar ASR accuracies.

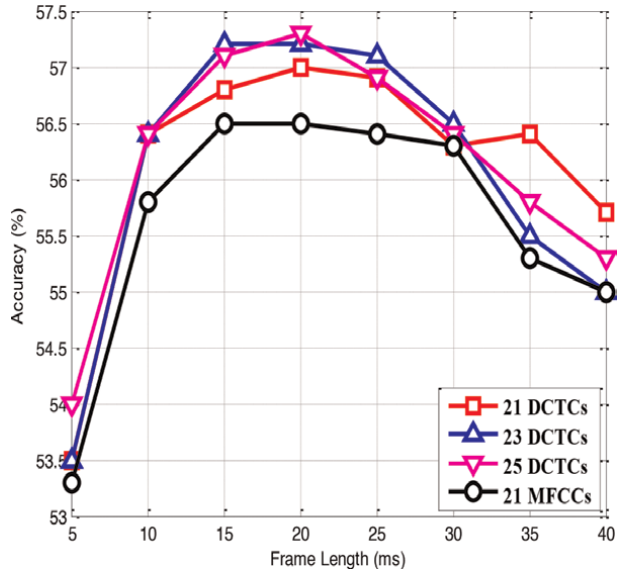

Figure 12.
Phone recognition accuracy as function of frame length using 21, 23, and 25 DCTCs.

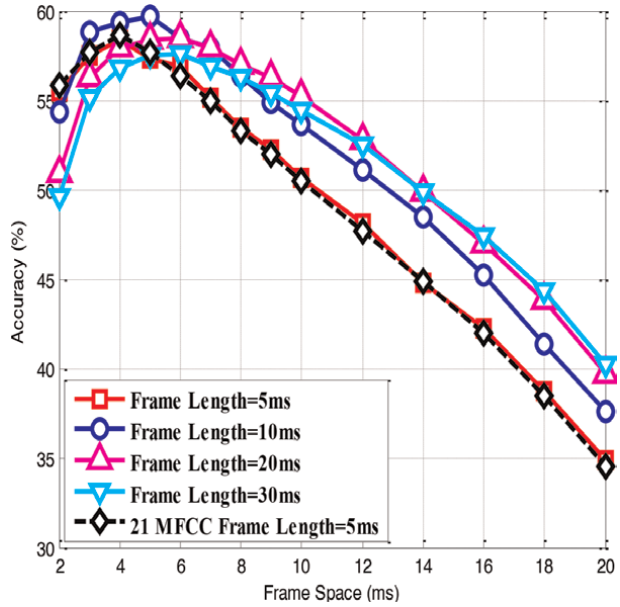

Figure 13.
Effect of frame length and frame space on phone recognition accuracy for 21 DCTCs.

Experiment A2-Time resolution effect for DCTCs: To investigate the role of time resolution on frame-based speech features, the feature "sampling rate" was varied by changing the frame spacing from 2 ms to 20 ms. Since the time resolution is also affected by frame length, 4 frame lengths (5, 10, 20, and 30 ms) were evaluated. 21 DCTCS were used for all tests. Other parameters were the same as for Experiment A1. The baseline in this experiment is the case of static MFCCs with frame length fixed at 5 ms. Results are shown in **Figure 13**.

Results vary from 34.9% (5 ms frames, 20 ms apart) to 59.7% (10 ms frames, 5 ms apart). Phonetic recognition accuracy degrades when the frame space is too large, especially for shorter frame lengths. The best performance for each frame length varies from 57.6% to 59.7%. As might be expected, the highest accuracy is obtained with short frame spaces and short frame lengths—that is high time resolution. However, unexpectedly, accuracy degrades when the frame space is too short. We hypothesize that oversampling of features is problematic for the HMM recognizer, due to the high correlation of features when frames are very closely-spaced.

Experiment A3—Effect of frequency warping on DCTC features: To evaluate frequency warping, bilinear frequency warping was used as in Eq. (7) with a single parameter α controlling the shape of the nonlinearity for the frequency warping. Bilinear warping with a coefficient of 0.45 closely approximates Mel warping, whereas a coefficient in the range of 0.5 to 0.57 approximates Bark warping [32].

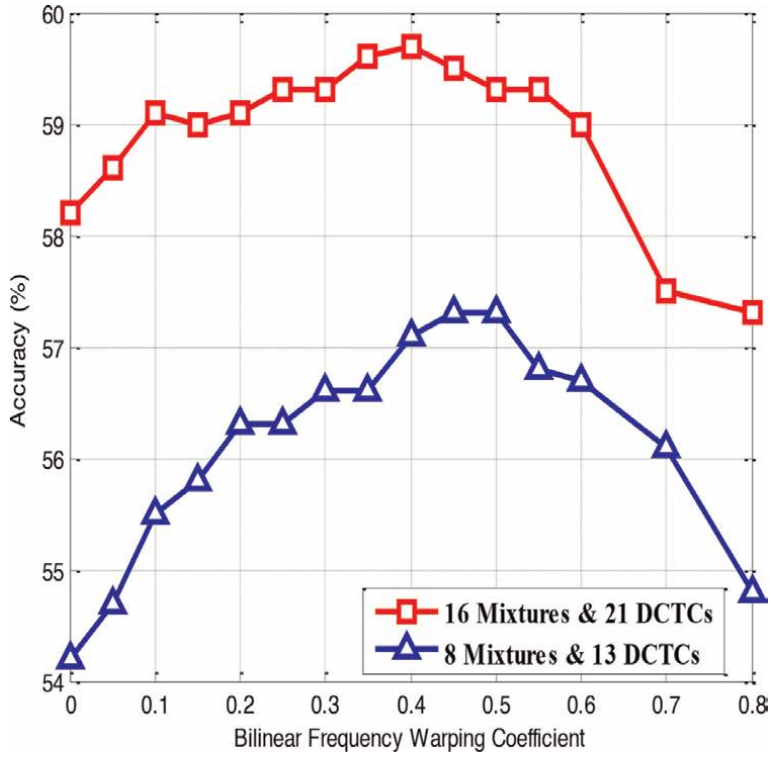

Figure 14.
Phone recognition accuracy as function of frequency warping for two cases: The standard Mel warping, i.e. g (f) = 2595log(1 + f/700), was used as a baseline, and results were within 0.1% of bilinear warping with a coefficient of 0.45 in both cases.

Since pilot experiments showed that the effects of frequency warping depend on the number of DCTC features and the number of HMM mixtures in the recognizer, these experiments were performed for two cases–13 DCTCs with 8 mixture HMMs; 21 DCTCs with 16 mixture HMMs. 10 ms frames spaced 5 ms apart were used in all cases. Results are plotted in **Figure 14** as the warping coefficient varies from 0 (no warping) to 0.8 (over warped).

The effect of warping is more apparent for the 8 mixture case than for the 16 mixture case. The overall best warping values found were.4 and.45 (most similar to Mel warping). For the 8 mixture/13 DCTC cases, the best warping of .45 resulted in a 3% accuracy improvement over the no warping case. For the 16 mixture/21 DCTC case, the best warping of .4 yielded a 1.5% accuracy improvement over the no warping case. The "standard" Mel warping, as proposed by O'Shaughnessy [31], was also evaluated as a baseline, and the result was within 0.1% of the result obtained using a bilinear warping coefficient of 0.45 for both 13 DCTCs/8 mixtures and 21 DCTCs/16 mixtures.

4.2.2 Experiment set 2: Dynamic features (DCTCs and DCSCs)

In these experiments, a myriad of parameters believed to be significant for DCTC/DCSC features which represent spectral-temporal characteristics in a block of frames centered on each frame were varied. These parameters include number of DCTCs/DCSCs, frame length/space, frequency/time-warping coefficients, and block length/space. Not all combinations of parameters were tested due to both the very large number of cases and the assumption that many of the variations would have much effect on ASR accuracy. Based on pilot experiments and the results reported previously for experiments B1, B2, and B3, many of these parameters were either fixed to a

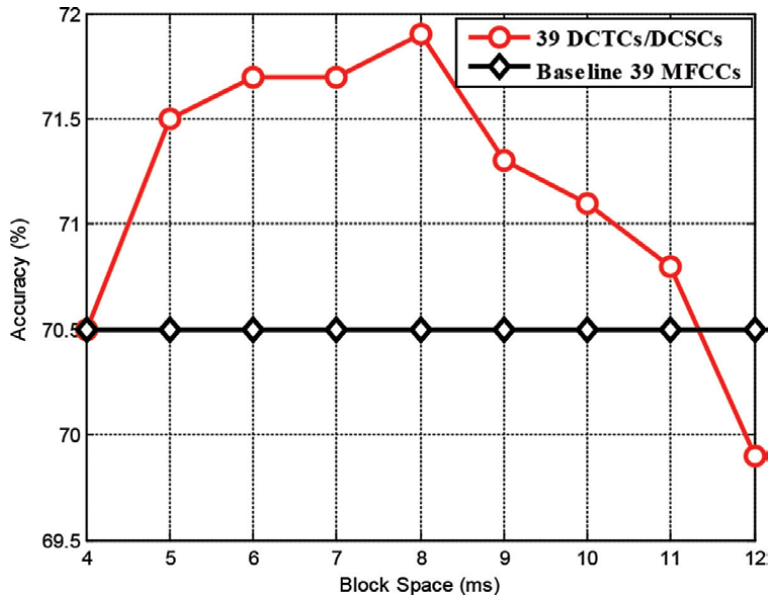

Figure 15.
Phone recognition accuracy of 39 DCTCs/DCSCs as function of block space with block length fixed at 251 ms: The 39 MFCC features produce a baseline of 70.5% (block space fixed at 8 ms).

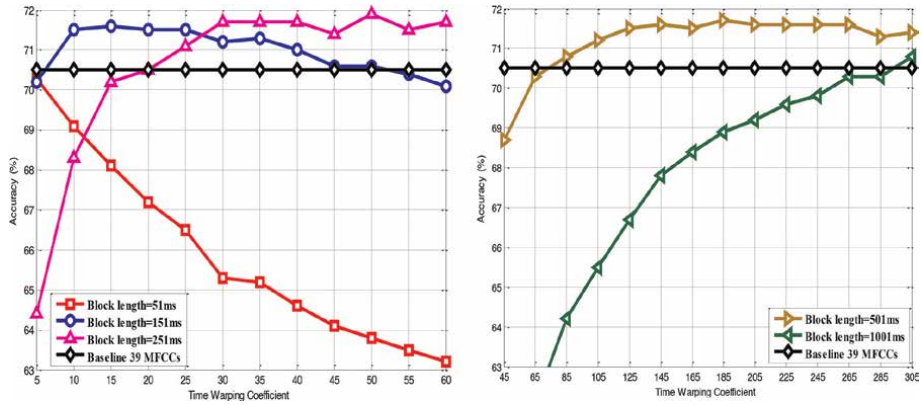

Figure 16.
Phone recognition accuracy as function of time-warping factor for different block lengths with a fixed block spacing of 8 ms: The baseline 39 MFCC case is also depicted.

single value or varied over a small range. The other parameters were varied and performance evaluated. 32 HMM mixtures were used due to the large dimensionality of the feature space.

Experiment B1—39 feature (13 DCTCs/3 DCSCs) experiments: Since 39 MFCC features are often used for ASR systems, the first experiments were performed with 39 features—13 DCTCs/3 DCSCs. The 39 MFCC feature case was the baseline. In this these experiments, the block length was fixed at 251 ms, and the effect of block spacing, which was also the feature "sampling" rate, was varied from 4 ms to 12 ms, with results depicted in **Figure 15**. The frame length/space was fixed at 10 ms/1 ms, and bilinear frequency warping (coefficient of 0.45) was used. The time-warping coefficient was 50 using a Kaiser window. The effect of block space on ASR accuracy varies approximately 2% from the lowest (12 ms) case to the highest (8 ms) case.

Experiment B2—39 features (13 DCTCs, 3 DCSCs), block length and time-warping effects (auditory time resolution): The objective of this experiment set was to examine the role of block length and time warping in representing the feature trajectories. These two parameters are closely related to the auditory time resolution of feature trajectories: A longer block length gives the ability to represent lower temporal modulation frequencies. A higher time-warping factor corresponds to higher time resolution in the central portion of a segment. To study these effects, five block lengths were used (51, 151, 251, 501, 1001 ms) with block spacing fixed at 8 ms. The time-warping factor of a Kaiser window was varied from 5 to 60 for the 51, 151, and 251 ms cases in steps of 5, and it was varied from 45 to 305 for the 501 and 1001 ms cases with steps of 20. The parameters for static features were identical to those in Experiment B1. Results are depicted in **Figure 16** again with a baseline of 39 MFCCs.

The highest accuracy of 71.9% was obtained with a time-warping coefficient of 50 using a block length of 251 ms. Results suggest that the block length and time warping are closely related to each other. As the block length increases, a larger time warping is required to achieve better performance, and a moderately long block length, such as 251 ms, which incorporates informative contextual information for each sample instant, provides the best result. However, very long contexts, such as 501 and 1001 ms, do not improve the performance and require very large-time-warping values. This shows that the spectral contexts too far from the current "observation point" do not provide much useful information, but can be suppressed by a large

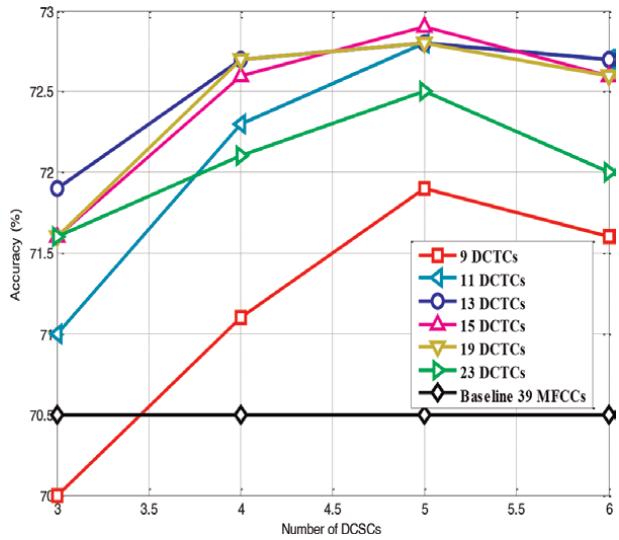

Figure 17.
Phone recognition accuracy as function of combinations of DCTCs and DCSCs.

time-warping factor, which emphasizes the useful information within a much shorter range surrounding the block center. Also, a long block length greatly increases the computations for each block (the number of multiplications in the vector inner product for computing the integration). Based on these considerations, 251 ms is considered the best value for the block length.

Experiment B3—Overall spectral-temporal effect: Phonetic recognition accuracy was evaluated with a variety of DCTC numbers (9 to 23 in steps of 2) and for a variety of DCSC numbers (3, 4, 5, and 6). These combinations were used to examine the trade-off between spectral and temporal resolution. Other parameters were selected to match the best parameter settings from earlier experiments (frame length/space of 10 ms/1 ms, bilinear frequency warping with coefficient of 0.4 for cases with 15 or more DCTCs and.45 for cases with fewer than 15 DCTCs, 251 ms/8 ms block length/space, and time warping of 50). Results are shown in **Figure 17**.

First, as the number of DCTCs increases beyond about 15, the performance begins to decrease. The number of DCSCs has a similar effect. Also, when a relatively small number of DCTCs were used, i.e. less overall spectral resolution, the performance increases relatively quickly as more DCSCs are used, i.e. more overall time resolution, as can be seen in the 9 and 11 DCTC cases (2% improvement from 3 DCSCs to 5 DCSCs using 9 DCTCs). However, the performance improves more slowly with more DCSCs when a relatively large number of DCTCs is deployed (less than 1% increment using 23 DCTCs). This observation shows the trade-off between the overall spectral and temporal resolution. The optimal "balance point" was obtained using 15 DCTCs and 5 DCSCs which produced 72.9% accuracy.

4.3 Independent EVAL set results and invariability

Based on the results from the TIMIT DEV set, a subset of parameters was further optimized. Two optimal parameter sets for a small feature set (27 features) and a large feature set (75 features) were obtained respectively. Also, the number of

GMM mixtures for each feature set was also optimized using the TIMIT DEV set. After these "final" optimizations were performed, to verify the generality of the tuned front end parameters, two EVAL phone recognition tasks were conducted with different data, as mentioned previously. The EVAL sets were the TIMIT EVAL set and the RASC863 Chinese Mandarin EVAL set. For the Chinese phone recognition task, the number of GMM mixtures was reduced due to the lower amount of available training data and the greater number of phone models to be trained. The best parameter values and the EVAL results are reported in **Tables 2** to **5**. In these tables, "BIG_REC" refers to the results using the optimal number of GMMs, indicating the best accuracy achieved by a high order HMM recognizer. In addition, the accuracy for the training set in each case is also reported, which shows an ideal upper bound of the recognizer performance if the training data completely represents the test data.

It can be seen from these results that the proposed DCTC/DCSC method achieves generally better performance than the baseline MFCC for independent EVAL sets. In addition, to further examine the feature invariability of the DCTC/DCSC front end, for the Chinese phone recognition task, the parameter values based on the TIMIT DEV set were varied and re-evaluated. These tests showed (results not given here) that the parameter values for best performance did not change, which meant that the parameter values determined from the TIMIT DEV applicable to an entirely different database in a vastly different language.

Experiment C1—DCTC/DCSC small feature set evaluation performance: The optimum settings for a small feature set are summarized in **Table 2**. Accuracies on the EVAL sets are reported in **Table 3**.

Experiment C2—DCTC/DCSC large feature set evaluation performance: The optimum settings for a large feature set are summarized in **Table 4**, and accuracies on the EVAL sets are reported in **Table 5**.

4.4 Unified framework explanation and statistical significance tests

As mentioned in Section 3 and in previous work [42], since the step of the amplitude scaling can be moved to immediately before the filterbank, the filterbank weights can be merged with the unwarped regular DCT basis vectors by a simple matrix product. Similarly, the delta and higher order acceleration dynamic terms can also be computed in a basis vector form. Thus, the proposed DCTC/DCSC front end and more

Parameter	Value
Frame Length	8 ms
Frame Spacing	1 ms
Frequency Warping $g(f)$	Bilinear, $\alpha = 0.45$
Number of DCTCs	9
Number of DCSCs	3
Frames per Block	251 ms (251 frames)
Block Spacing	7 ms (7 frames)
Time Warping (*dh/dt* term)	Kaiser window, $\beta = 50$

Table 2.
Optimum parameter settings for small feature set.

TIMIT Database	Number of HMM mixtures	EVAL Accuracy (%)	Training Accuracy (%)
Baseline 27 MFCCs	16	66.7	70.0
DCTC/DCSC	16	68.9	72.1
BIG_REC Baseline 27 MFCCs	80	69.2	79.2
BIG_REC DCTC/DCSC	80	71.3	81.2
RASC 863 Database	**Number of HMM mixtures**	**EVAL Accuracy (%)**	**Training Accuracy (%)**
Baseline 27 MFCCs	16	65.6	70.9
DCTC/DCSC	16	67.5	73.0
BIG_REC Baseline 27 MFCCs	48	68.8	79.1
BIG_REC DCTC/DCSC	48	70.4	80.6

Table 3.
27 feature TIMIT and RASC863 EVAL accuracies.

Parameter	Value
Frame Length	8 ms
Frame Spacing	1 ms
Frequency Warping $g(f)$	Bilinear, α= 0.4
Number of DCTCs	15
Number of DCSCs	5
Frames per Block	251 ms (251frames)
Block Spacing	7 ms (7 frames)
Time Warping (*dh/dt* term)	Kaiser window, β = 40

Table 4.
Optimum parameter settings for large feature set.

typically used filterbank front ends can be viewed as a unified framework. The reported experimental results can be explained using the unified time-frequency basis vectors as a common yardstick. First, Experiments A1 and A2 show that for static features, the proposed continuous Mel-shape warping results in slightly better performance than that obtained using Mel filterbank-derived basis vectors. By comparing their unified static basis vectors in **Figure 7(a)** and **Figure 10(a)**, our conjecture is that the quantization effect of the filterbank caused this difference. However, since the continuous Mel-shape warping and the filterbank are essentially two ways of implementing a Mel warping, the difference should be small as verified by the experimental results. It should be pointed out that it was experimentally verified that the standard way of implementing the MFFC front end, and MFFCs computed using unified basis vectors, result in identical feature values, provided the amplitude nonlinearity immediately follows the spectral magnitude step. Similarly, by comparing the unified dynamic basis vectors in **Figure 7(b)** and **Figure 10(b)**, it's clear that

TIMIT Database	**Number of HMM mixtures**	**EVAL Accuracy (%)**	**Training Accuracy (%)**
Baseline 39 MFCCs	32	69.7	76.2
DCTC/DCSC	32	72.5	79.4
BIG_REC Baseline 39 MFCCs	96	71.0	84.5
BIG_REC DCTC/DCSC	96	74.0	87.1
RASC863 Database	**Number of HMM mixtures**	**EVAL Accuracy (%)**	**Training Accuracy (%)**
Baseline 39 MFCCs	32	71.5	80.9
DCTC/DCSC	32	73.3	83.8
BIG_REC Baseline 39 MFCCs	64	72.0	85.0
BIG_REC DCTC/DCSC	64	74.2	87.1

Table 5.
75 feature TIMIT and RASC863 EVAL accuracies.

the non-uniform time resolution for a long segment of speech is a better representation of the spectral trajectory than the discrete time derivatives (most obvious in the zeroth order unified basis vectors). The more significant improvements over the baseline MFCC for various numbers of features in Experiments B and C support this observation.

Another set of experiments was conducted to address the issue of statistical significance. The goal was to show that the difference or similarity between the reported best cases for the DCTC/DCSC front end and the best baseline results in each previous experiment were statistically significant rather than due to noise or other random factors. These significance tests were conducted using the TIMIT database. To do this, the best results of the proposed method and the baseline were viewed as two random variables whose mean values were denoted as μ_T and μ_B. Then the 672 utterances of the TIMIT DEV and TIMIT EVAL sets were divided into 12 groups respectively, and test results were obtained for each group as samples. Since it's reasonable to assume the same (but unknown) variance for the proposed front end and the baseline

Experiment number	**Hypothesis tested**	**Results**
Exp. A1/A2, DEV set	$\mu_T > \mu_B$	Significant at 90% confidence level
Exp. A3, DEV set	$\mu_T = \mu_B$ (μ_T uses a warping of 0.45)	Significant at 97.5% confidence level
Exp. B1/B2, DEV set	$\mu_T - \mu_B \geq 1\%$	Significant at 90% confidence level
Exp. B3, DEV set	$\mu_T - \mu_B \geq 2.5\%$	Significant at 97.5% confidence level
Exp. C1 (both 16 and 80 mixtures, EVAL set)	$\mu_T - \mu_B \geq 2\%$	Significant at 90% confidence level
Exp. C2 (both 32 and 96 mixtures, EVAL set)	$\mu_T - \mu_B \geq 2.5\%$	Significant at 97.5% confidence level

Table 6.
Results of statistical significance tests for reported TIMIT experiments.

(because the database was identical in all cases), a t-test with 22 degrees of freedom was performed to test the significance of the difference term, i.e. $\mu_T - \mu_B$. The results of these tests are summarized in **Table 6**.

4.5 Frequency-dependent time warping in DCSC/DCTC scheme

In addition to the DCTC/DCSC implementation in which the time warping is independent of frequency, a slate of experiments for the DCSC/DCTC variation, which incorporated frequency-dependent time warping, was also conducted. The goal was to test the effectiveness of the auditory time-frequency trade-off caused by the nonlinear frequency selectivity for improving ASR performance. Specifically, the best warping factors obtained in the DCTC/DCSC experiments (i.e. 50 in the 27 feature case and 40 in the 75 feature case) were used as a baseline; smaller time warping for lower frequencies and larger time warping for higher frequencies were used with averages fixed at the baseline values (the block length was identical for all frequencies). Another equivalent method implemented was to use a longer block length for low frequencies compared to higher frequencies with the warping factor fixed. The results of these experiments showed no advantages over the baseline, which uses uniform time warping over all frequencies. This seems to imply that despite the results from human auditory research, which shows that humans have frequency-dependent temporal sensitivity [34–36], it may not play a crucial role, at least for the phone recognition ASR task evaluated in this chapter. Similar findings were observed by others. In one detailed study using wavelet signal processing to extract features for phonetic class recognition [51], the best performance obtained with wavelet features was only comparable to that obtained with MFCC features. In another study [52], a set of spectral-temporal features, which also accounts for the similar time-frequency trade-off, resulted in improved performance but only for restricted tasks (an isolated phone classification task rather than a continuous recognition application). The method introduced in [52] has not been adopted by the ASR community for general use.

5. Conclusion and future work

This chapter presents a generalized spectral-temporal feature extraction front end for representing speech information. The feature set is motivated by the attempt to mimic two primary properties of human hearing: frequency and time resolution. Based on a set of frequency and time-warping functions built into a set of modified 2-D cosine basis vectors and the trade-off between frequency and temporal and time resolution can be explored. A wide range of ASR experiments were conducted using the DCTC/DCSC method to comprehensively evaluate spectral-temporal resolution effects. This was done by adjusting parameters controlling the DCTC and DCSC parameters emphasize either spectral resolution or temporal resolution, and attempting to find the best overall "balance" point. The best combination point, using phonetic recognition experiments with the English language, also worked well with the Mandarin language.

Empowered by the front end unification approach, a higher level systematic unification can be envisioned. Conceptually, a recognizer front end should only require static features, with temporal patterns modeled by the recognizer. The human auditory system primarily performs spectral analysis whereas higher levels of

processing in the human brain appear to extract the longer terms spectral-temporal information. Apparently, the HMM framework is not able to adequately capture the temporal patterns contained in sequences of static speech features alone. Thus, it is possible that modeling of the "hidden" spectral-temporal patterns can be exploited by a data-driven training of a state-of-the-art recognizer, such as a deep neural network (DNN), which has the power of performing "deep learning."

Author details

Stephen A. Zahorian[1*], Xiaoyu Liu[2] and Roozbeh Sadeghian[3]

1 Binghamton University, Binghamton, USA

2 Dolby Laboratories Inc., San Francisco, USA

3 Harrisburg University of Science and Technology, Harrisburg, USA

*Address all correspondence to: stephen.zahorian16@gmail.com

References

[1] Zahorian SA. Detailed Phonetic Labeling of Multi-Language Database for Spoken Language Processing Applications. Rome, NY, USA: Air Force Research Laboratory Information Directorate; 2015. Available from: http://www.oracle.com/us/corporate/citizenship/corporate-citizenship-report-2563684.pdf. DOI: 10.21236/ada614725

[2] Peterson GE, Barney HL. Control methods used in a study of the vowels. The Journal of the Acoustical Society of America. 1952;**24**(2):175-184. DOI: 10.1121/1.1906875

[3] Hermansky H. Perceptual linear prediction analysis of speech. The Journal of the Acoustical Society of America. 1990;**87**(4):1738-1752. DOI: 10.1121/1.399423

[4] Weber K, Wet F, Cranen B, Bodes L, Bengio S, Bourlard H. Evaluation of formant-like features for ASR. Int. Conf. on Spoken Language (ICSLP). 2002. DOI: 10.1121/1.1781620

[5] Garner P, Holmes W. On the robust incorporation of formant features into hidden Markov models for automatic speech recognition. Proceedings of ICASSP. 1998:1-4. DOI: 10.1109/ICASSP.1998.674352

[6] Holmes J, Holmes W, Garner P. Using formant frequencies in speech recognition. Proceedings of EUROSPEECH'97. 1997;**4**:2083-2086

[7] Bogert BP, Healy MJR, Tukey JW. The quefrency analysis of time series for echoes: Cepstrum, pseudo autocovariance, cross-cepstrum and Saphe cracking. In: Rosenblatt M, editor. Chapter 15. Proceedings of the Symposium on Time Series Analysis. New York: Wiley; 1963. pp. 209-2243

[8] Zwicker E, Fastl H. Chapter 3. In: Psychoacoustics, Facts and Models. Springer-Verlag; 1990. pp. 25-28

[9] Stevens SS, Volkmann J, Newman EB. A scale for the measurement of the psychological magnitude pitch. The Journal of the Acoustical Society of America. 1937;**8**(3):185-190. DOI: 10.1121/1.1915893

[10] Fletcher H. Auditory patterns. Reviews of Modern Physics. 1940:12

[11] Zwicker E. Subdivision of the audible frequency range into critical bands. The Journal of the Acoustical Society of America. 1961;**33**(2):248-248. DOI: 10.1121/1.1908630

[12] Glasberg BR, Moore BCJ. Derivation of auditory filter shapes from notched-noise data. Hearing Research. 1990;**47**(1–2):103-138. DOI: 10.1016/0378-5955(90)90170-T

[13] Bridle JS, Brown MD. An Experimental Automatic Word-Recognition System. JSRU Report. Vol. 1003. Ruislip, England: Joint Speech Research Unit; 1974

[14] Patterson PD, Robinson K, Holdsworth J, McKeown D, Zhang C, Allerhand MH. Complex sounds and auditory images. In: Cazals Y, Demany L, Horner K, editors. Auditory and Perception. Oxford, UK: Pergamon Press; 1992. pp. 429-446

[15] Slaney M. An Efficient Implementation of the Patterson-Holdsworth Auditory Filter Bank. Apple Technical Report. Cupertino, CA: Advanced Technology Group, Apple Computer, Inc.; 1993. p. 35

[16] Zhang X, Heinz MG, Bruce IC, Carney LH. A phenomenological model for the response of auditory-nerve fibers: I. nonlinear tuning with compression and suppression. The Journal of the Acoustical Society of America. 2001; **109**(2):648-670. DOI: 10.1121/1.1336503

[17] Robinson DW, Dadson RS. A redetermination of the equal-loudness relations for pure tones. British Journal of Applied Physiology. 1956;**7**:166-181

[18] Makhoul J. Linear prediction: A tutorial review. Proceedings of the IEEE. 1975;**63**:561-580. DOI: 10.1109/PROC.1975.9792

[19] Memon S, Lech M, Maddage N. Speaker verification based on different vector quantization techniques with Gaussian mixture models. In: Third Int. Conf. On Network and System Security. 2009. pp. 403-408. DOI: 10.1109/NSS.2009.19

[20] Jayanna HS, Prasanna SRM. Fuzzy vector quantization for speaker recognition under limited data conditions. TENCON 2008-IEEE Region 10 Conference. 2008:1-4. DOI: 10.1109/TENCON.2008.4766453

[21] Chen J, Paliwal KK, Mizumachi M, Nakamur S. Robust MFCCs Derived from Different Power Spectrum. Scandinavia: Eurospeech; 2001

[22] Wang C, Miao Z, Meng X. Differential MFCC and vector quantization used for real-time speaker recognition system. IEEE Congress on Image and Signal Processing. 2008: 319-323. DOI: 10.1109/CISP.2008.492

[23] Drullman R, Festen JM, Plomp R. Effect of reducing slow temporal modulations on speech reception. The Journal of the Acoustical Society of America. 1994;**95**(5):2670-2680. DOI: 10.1121/1.409836

[24] Athineos M, Hermansky H, Ellis DPW. LPTRAPS: Linear predictive temporal patterns. In: Proc. of Interspeech. Jeju Island, Korea; 2004. pp. 1154-1157

[25] Valente F, Hermansky H. Hierarchical and parallel processing of modulation spectrum for ASR applications. ICASSP. 2008:4165-4168. DOI: 10.1109/ICASSP.2008.4518572

[26] Kleinschmidt M. Methods for capturing spectro-temporal modulations in automatic speech recognition. Acustica united with acta Acustica. 2002;**88**:416-422

[27] Kleinshmidt M. Localized Spectro-Temporal Features for Automatic Speech Recognition. Switzerland: Eurospeech; 2003

[28] Allen J. Short term spectral analysis, synthesis, and modification by discrete Fourier transform. IEEE Trans. Acoust., Speech, and Signal Processing. 1977; **ASSP-25**(3):235-238. DOI: 10.1109/TASSP.1977.1163007

[29] Kim C, Stern RM. Feature extraction for robust speech recognition using a power-law nonlinearity and power-bias subtraction. INTERSPEECH. 2009:28-31. DOI: 10.21437/Interspeech.2009-5

[30] Rao KR, Yip P. Discrete Cosine Transform: Algorithms. Academic Press; 1990. Advantages. Applications

[31] O'Shaughnessy D. Speech Communication: Human and Machine. Addison-Wesley; 1987. p. 150

[32] Smith JO, Abel JS. The bark bilinear transform. In: Proceedings of the IEEE Workshop on Applications of Signal

Processing to Audio and Acoustics. New York; 1995. DOI: 10.1109/ASPAA.1995.482991

[33] Wang S, Sekey A, Gersho A. An objective measure for predicting subjective quality of speech coders. IEEE Journal on Selected Areas in Communications. 1992;**10**(5):819-829. DOI: 10.1109/49.138987

[34] Duifhuis H. Consequences of peripheral filter selectivity for nonsimultaneous masking. The Journal of the Acoustical Society of America. 1973;**54**(6):1471-1488

[35] Bidelman GM, Khaja AS. Spectrotemporal resolution tradeoff in auditory processing as revealed by human auditory brainstem responses and psychophysical indices. Neuroscience Letters. 2014;**572**:53-57

[36] Shailer MJ, Moore BCJ. Gap detection as a function of frequency, bandwidth, and level. The Journal of the Acoustical Society of America. 1983; **74**(2):467-473. DOI: 10.1121/1.389812

[37] Meyer B, Ravuri SV, Schadler MR, Morgan N. Comparing different flavors of spectro-temporal features for ASR. INTERSPEECH. 2011:1269-1272. DOI: 10.21437/Interspeech.2011-103

[38] Depireux DA, Simon JZ, Klein DJ, Shamma SA. Spectro-temporal response field characterization with dynamic ripples in ferret primary auditory cortex. Journal of Neurophysiology. 2001;**85**: 1220-1234. DOI: 10.1152/jn.2001.85.3.1220

[39] Ge W. Two Modified Methods of Feature Extraction for Automatic Speech Recognition (Thesis). Binghamton: Department of Electrical and Computer Engineering, Binghamton University; 2013

[40] Hermansky H, Morgan N. RASTA processing of speech. IEEE Trans. Speech and Audio Processing. 1994;**2**(4): 578-589. DOI: 10.1109/89.326616

[41] Hermansky H, Sharma S. TRAPS - classifiers of temporal patterns. ICSLP. 1998;**3**:1003-1006

[42] Liu X, Zahorian SA. A Unified Framework for Filterbank and Time-Frequency Basis Vectors in ASR Front Ends. Australia: ICASSP; 2015. DOI: 10.1109/ICASSP.2015.7178854

[43] Chiu BY, Bhiksha R, Stern RM. Towards fusion of feature extraction and acoustic model training: A top down process for speech recognition. INTERSPEECH. 2009:32-35. DOI: 10.21437/Interspeech.2009-6

[44] Chiu BY, Stern RM. Analysis of physiologically-motivated signal processing for robust speech recognition. INTERSPEECH. 2008:1000-1003. DOI: 10.21437/Interspeech.2008-291

[45] Young S et al. The HTK Book (for HTK Version 3.4) Available from: http://htk.eng.cam.ac.uk/. Cambridge University; 2009 Revised for HTK Version 3.4

[46] Zue V, Seneff S, Glass J. Speech database development at MIT: TIMIT and beyond. Speech Communication. 1990;**9**:351-356. DOI: 10.1016/0167-6393(90)90010-7

[47] Lee K, Hon H. Speaker-independent phone recognition using Hidden Markov Models. IEEE Trans. on Acoust., Speech, and Signal Processing. 1989;**37**(11): 1642-1648. DOI: 10.1109/29.46546

[48] Li A, Yin Z, Wang T, Fang Q, Hu F. RASC863-a Chinese speech corpus with four regional accents. Report of Chinese Academy of Sciences. 2004

[49] Nossair ZB, Silsbee PL, Zahorian SA. Signal Modeling Enhancement for Automatic Speech Recognition. Vol. 1. Proceedings of ICASSP; 1995. pp. 824-827. DOI: 10.1109/ICASSP.1995.479821

[50] Zahorian SA, Wong B. Spectral amplitude nonlinearities for improved noise robustness of spectral features for use in automatic speech recognition. The Journal of the Acoustical Society of America. 2011;**130**(4):2524. DOI: 10.1121/1.3655077

[51] Van Pham T. Wavelet analysis for robust speech processing and applications (thesis). Graz University of Technology. 2007

[52] Droppo JG III. Time-frequency features for speech recognition (thesis). University of Washington. 2000

Section 3

Classical Studies

Chapter 5

Perspective Chapter: Difficulties for Translating Quevedo's Sonnets from Portuguese Translations into English

Leonor Scliar-Cabral

Abstract

The aim of this chapter is a discussion about the criteria used on the translation of Brazilian Portuguese poetry into American English. I thus exemplify it with the translations of Quevedo's sonnets, namely the sonnet "*Desde a torre*," "From the tower." As one goes through Quevedo's sonnets, one can notice the recurrence of the semantic fields "fire" and "prison." The first appears until fatigue in the opposites game within the Petrarquean pairs tradition like fire/snow~water. In order to enjoy the multiple readings that Quevedo offers, it is necessary to delve into the disappointments of which he was a victim and the seventeenth-century Spain collapse, a tumultuous scene of the Baroque: Spain was ravaged by the Thirty Years' War and defeated by Holland and France. I based the criteria upon translation theories, more specifically upon poetic translation. I describe in detail the genre sonnet, particularly its metrics and rhymes, and the difficulties of translating them. I end the chapter with a microanalysis of the poem "*Desde a torre*" translation into American English.

Keywords: poetry translation, Brazilian Portuguese, American English, Quevedo's sonnet, Baroque

1. Introduction

In order to enjoy the multiple readings that Quevedo (Francisco Gómez de Quevedo y Villegas, 1580–1645) offers, it is necessary to delve into the disappointments of which he was a victim and the seventeenth-century Spain collapse, a tumultuous scene of the Baroque: Spain was ravaged by the Thirty Years' War and defeated by Holland and France. It is against this backdrop that the Spanish literary Baroque thrives, whose main characteristics are cultism and conceptism, with Quevedo being an adept of the latter, unlike Gôngora.

The sonnet's author considered the most beautiful in the Spanish language by Alonso [1], "*Cerrar podrá mis ojos la postrera/ sombra*," "My eyes will close the ultimate, and last/shadow," was exiled three times due to palace intrigues, and it is none other than one of these that inspired the title of the sonnet "From the tower."

Mentioning the topos versed by him, from the satirical to the amorous, it is undoubtedly when dealing with death and life brevity that conceptism is handled most vigorously: neostoicism bases his ideas, in the wake of Seneca's thought, revisited by Christianity. Conceptism must be understood as a style that can be either epigrammatic or vigorous, whose concise words or phrases shine with the force and speed of lightning. There is a lot of wordplay and lively metaphors. But Quevedo's characteristic touch, as Alonso points out [1] in several steps, is his affectivity: the "eruptive vitality," the "forging fury," and the "condensation."

Quevedo is, perhaps, among the poets whom I translated [2], the most Hispanic without, however, ceasing to be the most universal and, at the same time, historical: imprisoned emotions that finally explode, the satirization of human villainies, the panel of a Spain in decadence are portrayed with the formal domain of someone who masters Petrarque's technique, Mannerism and Baroque, although he does not allow himself to be subjugated by them.

The imprisoned passions idea has its source more in Quevedo's existential vision than perhaps in experiences with real women, but he had a very concrete prison experience three times. In the last arrest, at the end of his life, from 1639 to 1643, he remained under house arrest, under suspicion of political satire against King Filipe IV (poor whoever fell under Quevedo's sights!).

2. Quevedo's topos

With a vast work, whose chronology has not yet been established with precision, the theme that stands out in the sonnets I have translated [2] is the explosion of affections.

As one goes through Quevedo's sonnets, one can notice the recurrence of the semantic fields "fire" and "prison." The first appears until fatigue in the opposites game within the Petrarquean pairs tradition like fire/snow ~ water, which in the collection I translated [2] is illustrated in the sonnets: "*Cerrar podrá mis ojos la postrera*," "My eyes will close the ultimate, and last"; "*En los claustros del alma la herid*a," "In the cloisters of soul the suffered wound."

But the obsession with the inner fire also appears in his lexical and metaphorical preferences by terms such as "marrows" (in "My eyes will close the ultimate, and last, "verse 11); (in "In the cloisters of soul the suffered wound," verse 4).

A Bosch of literature, Quevedo hyperbolically emphasizes the human grotesque, illustrated below with the sonnet that satirizes female old age.[1] It's exactly on the satirical side where Quevedo presents one of his great originalities, drinking in us popular records, he populates his verses with Spanish words and expressions that were in people's mouths, such as *antaño* (yesteryear), *morenos* (dark skinned), and *présteme un ochavo* (lend me an ochavo), contradicting the prevailing trend for cultisms.

On the other hand, a true panel of Hispanic decadence, where his preference for the oppressed is clear, parades before us, this time reminiscent of Goya. And not least, he was the victim of inquisitorial processes, for mocking hell itself. His appeals for death to welcome him reach the maximum lyricism in the sonnet "Already great and formidable sounds" (**Figure 1**):

Many times, almost translating the authors within this cultural tradition (which does not diminish its literary value, as the great contributions of Plautus and Terence are examples), Quevedo presents originalities, as I have already pointed out: his pagan-Christian

[1] A topic dear to Greek and Latin epigramists, cf. the Palatine Anthology, book XI, [3].

Se um feliz descanso, paz serena,	If pleasant rests, a comfortable peace
paramentada em dor a morte envia,	with rich vestments in pain death sends to us,
finde-me a vida e meu viver intime.	finish my life and end it subpoena.

Figure 1.
"Se um feliz descanso" into English.

syncretism, whose inspiration can be traced back to the stoicism of Seneca's or to Marcus Aurelius' theme of life fleetingness, never obscures the strongest side of his literary production, which is to express the human passions and the formal resources core: they are not an end in themselves, but create tensions, which mirror the most intimate energies, starting with syntactic parallels and forming a type of correlation.

3. Is poetic translation possible?

The debate on poetry translatability is inserted, firstly, in violating the original text of translation in general. According to Paes [4] "From its earliest beginnings, translation can thus be seen as an effort to make the positive impulse of language towards openness and communication prevail over the negativity of its other impulse towards closure and communication exclusion" [...] "the translator is a builder of linguistic bridges that link the closed-in-itself idiomatic islands to one another."

Paes [4] is, therefore, among those, who admit translatability, citing Walter Benjamin [5], when referring to the restoration in human memory of the "traces of the Adamic language erased by Babel curse": the existence of a Universal human cognition, despite linguistic diversity, would support translatability, since in the translation process the translator starts with the text interpretation in the source language and concludes his work, providing the receiver with the same interpretation in the target text.

According to Gadamer [6] "The meaning must be maintained, but because it has to be understood in a new linguistic world, it will come into force in a new way. That is why all translation is interpretation," an idea shared by Theodor [7]. Further on, Gadamer [6] emphasizes the role of recreation in the translation process.

In the opposite direction, there are several translation theorists, whose epistemological basis is in line with the linguistic relativism of Sapir [8] and Whorf [9], that linguistic systems, conditioned spatially and temporally, shape the way of perceiving reality and, therefore, of thinking and this would make it impossible to translate different worldviews.

In a period (nineteenth century beginning), when translation still focuses heavily on literal, word-for-word translation, Wilhelm von Humboldt [10] states in the preface to his translation of Aeschylus' *Agamemnon*: "Analysis and experience confirm that, which has been observed more than once: apart from expressions, which designate only physical objects, no word in one language is perfectly the same as one in another."

Coseriu [11] criticizes the word-for-word approach to literal translation, asserting that "only texts are translated," taking into account, in addition, the various extralinguistic contexts involved: empirical, historical, cultural, etc.

Paes [4], however, while accepting translatability, as I commented above, with regard to poetry, states: "In no other sector of translation activity is the struggle against the isolationist impulse of language more fierce than in poetry translation" [...] "This insofar as, to mean the most with the least, it makes use of all language

expressive resources, largely peculiar, exclusive to it" [...] "To the despair of the poetry translator, the phonic extract is always the most idiosyncratic" [...] "the poetry translator is sometimes led to more radical solutions than those commonly adopted in the routine of translating. This can creatively influence the very use of the language into which he translates, enriching it with resources of 'strangeness'." However, Humboldt [10] warns: "Insofar as it makes the strangeness feel rather than the strangeness, translation has achieved its highest ends, however, the moment the strangeness itself appears, perhaps even obscuring the strange, the translator reveals not to be up to his original."

Humboldt [10] also recognizes the impossibility of translating poetry, when referring to *Agamemnon*. The uniqueness of each message is linked to the concept of energy: "each one must carry within himself his work and that of all the others, this emergence should resemble the emergence of an ideal figure in the artist's imagination. Nor can it be extracted from something real, it arises by a pure spirit energy, and rather out of nothing, but from that moment on it starts to live and be real and lasting."

A very strong line of poetic translation defends the idea that only a poet is competent to do it. The arguments rest on the eminently creative character of poetic translation with all its implications for the permissible audacity of violating the original text according to Paz [12] and Islam [13].

Eco [14] points out that the full meaning of the poetic text also rests on the phonic suggestions, as well as on the rhyme and other sound aspects, such as paronomasia, alluding to Jakobson's analysis of the example "I like Ike." It is necessary to emphasize that Jakobson [15] was not referring strictly to poetry, but rather to language poetic function, which he defined as having the message itself as its focus (also for Silvestri [16]. Thus, the poetic function may be the predominant one in the girl's choice of the adjective "horrible," in the expression "The horrible Harry," instead of "dreadful," "terrible," "frightful," or "disgusting," because it is paronomasia. Eco [14] concludes that the notion of propositional content only applies to utterances that unambiguously represent states of the world and never to those, in which the focus is the message itself, as occurs when the poetic function predominates. The expressive substance, therefore, for Eco [14] is "fundamental with regard to phono-stylistic topics and discursive rhythm in general." But it is precisely on the expressive substance that many deny the possibility of translation, as for instance, Silvestri [16].

The difficulties in translating poetry and artistic prose had already been pointed out by Schleiermacher [17], for whom "the language musical element that manifests itself in rhythm and intonation is also of special and superior importance" [...] "Therefore, what strikes the sensitive reader of the original work in this respect as characteristic, intentional and effective in tone and mood terms, and as decisive for the speech rhythmic or musical accompaniment must also be transmitted by the translator."

In an intermediate position, there are theorists such as Dollerup [18], who claim that it is not possible to find equivalences in poetic translation, but rather compensatory strategies, such as the use of traits considered poetic by a given culture.

The concept of seeking equivalences in the target language has been one of the most debated by translation theorists. Schleiermacher [17] in his classic essay on the different methods of translation, when commenting on literary and scientific translation difficulties, points out the difference between "equivalent expression," which is impossible and "closer" one, which is possible. On the other hand, Coseriu [11], based on his tripartite model of Meaning 1 (the meaning of the source language), Meaning 2 (the meaning in the target language), and Designation, that is, the referent, only

admits equivalences in the designation. For him, "in the translation process itself – it is about finding meanings in the target language that can designate the same thing." In fact, Coseriu [11] argues that linguistic contents are in an irrational relationship, which he calls incommensurability: in his view, it is the main problem of translation theory, that is, "the problem of identical designation," with different linguistic means" [11]. However, the identical designation collides with the differences between cultures, spatially and temporally distanced, as ironically emphasizes Nietzsche [19]: "There are translations with honest intentions that are almost falsifications and involuntary vulgarizations of the original, only because their cheerful and courageous time could not be translated – time which, providentially omitted, overcomes all that is dangerous in things and words."

When dealing with the issue of incommensurability, Eco [14] asserts that, although it is a fact, it cannot be equated with incomparability. The possibility of comparing broadens the translatability horizons, as long as the translator is not restricted to linguistic limits, making use of intertextuality, narrativity, and psychological aspects.

Eco [14] approaches the issue from another perspective, by proposing the existence of a deep meaning in the texts, to which the translator has access through interpretation, superficializing it, when translating it into the target language, even at the cost of lexical and referential violations.

When stating that translations "are not about linguistic types, but about linguistic occurrences" and, when recalling the Chomskyan dichotomy of deep and superficial structures, several implications follow: on the deep meaning, superficialized in the utterances, several factors intervene, which Coseriu [11] calls contexts.

A first factor that Eco [14] mentions is the total work context, exemplified by *al di là dela siepe* translation of his book *Il pendolo di Foucault* (Foucault's Pendulum). It is a literary quotation from the sonnet *L'infinito* by Giacomo Leopardi, like hundreds of others that fill the book, to highlight the style of the three characters. The English translator Weaver did the best translation, with an "explicit reference to Keats": "we glimpsed endless vistas." Another factor pointed out by Eco is the cultural differences that interfere in the translation of even banal situations. The French expression "*Cherchez la femme,*" cannot be translated as "Look for the woman," because its meaning is "where there is trouble, the cause is in some woman" [14].

4. The need for poetic translation

Despite the immense difficulties that poetic translation faces at, it is necessary. Several theorists point out that it allows those, who do not know the source language, to have access to new art forms and new worldviews, in addition to enriching the target language with new forms of expression. Humboldt [10] cites the example of German meter enrichment, after the Greek classics' translation by Klopstock. I cite the contribution of Plautus and Terence, when they translated the Greek New Comedy and, thus, established the Latin meter.

The enrichment of the target language and culture is not only provided by poetic translation: the impact on the German language and culture with the Bible translation into the vernacular by Luther was pointed out by Humboldt [10]; Eco [14] mentions the radical change in the style of the French philosophical genre, after Heidegger's translation; the same happened with the Italian narratives, after the American authors' translation, before the Second World War. As a result, according to Eco, the

axis of the debate shifts from the relationship between source and target language to the "effect that the translated text has on the target culture."

5. Difficulties in poetic translation

Translating poetry is a challenge, as it implies finding the stylistic resources in the target language, that is, finding, among the parameters that define a given genre, those that will cause on the reader a similar effect to those felt by the readers of the source text.

Borges [20] confessed that if he could translate music from English or German into Spanish, he would be a great poet. Paraphrasing, it can be said that only those who deal with musical effects can translate poetry and this is the biggest challenge, that is, reconciling the meanings intended by the author with aesthetic solutions, finding their aesthetic availability in the target language, or, as stated by Dámaso Alonso [1], "in poetry there is always a motivated link between signifier and signified."

Given the difficulties in finding equivalences in the target language, particularly with regard to poetic translation, theorists have suggested compensatory strategies to obtain similar effects during reception. Gadamer [6] explains the sometime painful path, in which the translator seeks a middle ground to reconcile the interpretation he makes of the source language text in order to make it available in the target language, through back and forth.

Such a middle ground is also suggested by Goethe [21] when he explains that there are two maxims in translation, the first one being to bring the culture from which the text to be translated comes to the culture of the target language, with an incorporation, while the other consists of an inverse trajectory, that is, subjection "to the conditions, their way of speaking, their particularities." When in doubt, Goethe suggests the first option.

Schleiermacher [17] conceives the ideal that translation would be, either enabling the reader to meet the author or conversely, for the author to meet the reader. In the second hypothesis, the author would be able to write the work in the target language, which would be the perfect translation. It is surprising that in 1813 Schleiermacher already mentioned "We could outline rules for each of the two methods, taking into account the different genres of discourse" [17].

To get an idea of the difficulties in translating a Petrarquean sonnet, such as the sonnet "From the tower," I will begin by examining the questions of meter and rhyme. To solve them, strategies are used that involve inversions, additions, omissions, and substitutions of items, preferably purely grammatical ones, although additions and losses of items and/or expressions that refer to external meaning are also used; major displacements generally imply morphosyntactic changes.

In the sphere of meaning, resources are used, such as the use of items belonging to the same semantic field and figures as metaphors. Sometimes, some more audacious resources are used, interpreted as the space of freedom left to the poetic creation of the translator.

Such difficulties increase when the differences between the source language and the target language are compared, as it occurs in the translation from Portuguese into English. The following are systematized and further elaborated on:

a. American English (AE) has almost 13 oral vowels, including lax and tense ones, unlike Brazilian Portuguese (BP), which has seven oral vowels and five nasal

ones. The contrast between syllables in words, in BP, is marked by the stressed and unstressed syllables, while in AE, it is between lax and tense syllables, which influences the rhythm in both languages;

b. BP has no affricate consonants, nor dental or glottal fricatives;

c. Phonotactic rules differ mainly because BP does not admit plosive or nasal consonants at the end of a syllable, therefore, at the end of a word;

d. The syntactic rules are also different, as the AE does not admit the subject postponed to the verb in declarative affirmative sentences, nor the postposition of the adjective to the noun, as well as it does not admit the ellitic or null subject;

e. there are many differences in the use of the verbal aspect;

f. false cognates.

6. More serious rhyming problems

The most serious rhyming issues in translation from BP into AE stem from differences between the two phonological systems, particularly with regard to the vowel system, since, as seen, BP has five nasal vowels and AE does not have any although vowel letters in the sonnets are the same, as, a, e, i, o, u, since both written languages adopt the same script, the Latin one. There are also differences regarding nasal consonants that, in English, can appear in syllabic locking, unlike Portuguese, which compensates for this gap with nasalized diphthongs, which are absent in English.

Failure to observe these differences caused several stumbling blocks in the translation by Horta, Vianna and Rivera [22], as shown in the following example, in Cristóbal de Castillejo's sonnet "*Sonho*" ("Dreaming") the translators rhyme "cousa" (/ˊkowza/), with "saborosa" (/saboˊrɔza/), and "formosa" (/foRˊmɔza/).

This constitutes a trap for translating the rhymes, because, although the letters are the same, the grapheme values depend on the phonological system of each language.

Meter problems, particularly in heroic verses, that is, decasyllables with ictus on the sixth foot, stem from many factors, firstly, morphosyntactic issues, such as: the AE does not admit the subject postponed to the verb in declarative affirmative sentences, nor the postposition of the adjective to the noun, as well as it does not admit the ellitic or null subject; secondly, prosody questions, since the contrast between syllables in words, in BP, is marked by the stressed and unstressed syllables, while in AE, it is between lax and tense syllables. Therefore, as I stated above, several strategies are necessary that I will specify below:

a. inversions, additions, omissions, and substitutions

Example of inversion, addition, and substitution in the translation of the sonnet "From the tower," I found in the fourth verse of the first stanza (**Figure 2**):

b. another strategy I will comment on deals with semantics and figures, particularly metaphors. It must be assumed that approximate meanings can be found in the source language and in the target one but never the senses, since their respective

e os mortos eu escuto, olhos despertos.	I hear the deads, the eyes vigilant, open.

Figure 2.
"e os mortos eu escuto" into English.

Alma que todo um deus a tem rendido,	Soul, that a whole god has surrendered,
veias que a tanto fogo humor têm dado,	veins, that have given much humor to fire,
medulas que gloriosas têm ardido,	marrows that glorious have become burned,
	your entire body will leave, not your care;
seu corpo deixarão, não seu cuidado;	
serão cinza, terão, porém, sentido:	they will be gray, but they will be followed:
pó serão, porém, pó enamorado.	they will be dust, but enamoured powder.

Figure 3.
"Alma que todo um deus" into English.

readers do not share the same sociocultural experiences, with spatiotemporal coordinates being implicit. Such a difference already allows the literary translator a wide margin of creativity, without too much discrepancy from the original meanings. Therefore, he will use resources such as replacing words with items belonging to the same semantic field, paraphrases, and tropos. Sometimes the translator eliminates items not essential to the central idea.

c. The last strategy deals with syntactic parallelism, exemplified in the bellow sonnet "The eyes will close the last." There are three subjects of split clauses, asyndetically coordinated, each followed by an adjective clause (in the first tercet), creating a tension that is solved in the last stanza, with the implicit resumption of the three vocatives, first, in an asyndetically coordinated clause, then in a coordinated adversative clause, and finally, in a clause in which only the predicate is coordinated by the adversative (**Figure 3**):

7. The sonnet

The sonnet is a poetic form that dates from the thirteenth century, having originated in Sicily, at the court of Frederick II, and coexists with Provence poetry, but the poet Guittone D'Arezzo was the one who established its form. The first great classics of the genre are Petrarch and Dante. The first bequeathed its name to one of the most widespread sonnet forms, the Petrarquean sonnet, adopted by Camões. Shakespeare composed his sonnets in the English form, with three quartets and a couplet. In Portugal, Sá de Miranda introduced the sonnet and several other poetic genres, constituting the so-called *dolce stil nuovo*, after having made a trip to Italy in the first quarter of the fifteenth century.

The structure adopted in Quevedo's sonnets is Petrarchean. Despite being criticized, the sonnet continues to be widely practiced. The great Brazilian poet Francisco

Carvalho [23] asks: "After all, if the sonnet is really out of fashion, outdated in form and content, why do so many people continue to write it with such conviction?"

I will begin by explaining the units of which the verse is constituted in the sonnet. They are the feet (probably a metonymy derived from the rhythm marking), and they have their origin in the Greek lyric, later adopted by the Romans. Plautus and Terence were the Latin meter creators, when they translated the New Greek Comedy. We can therefore conclude that the translator's role goes far beyond translation. In both Greek and Latin, the phonological accent is based on duration, as it is the case in English, and therefore, so it is the meter. Thus, what counts is how the long and short syllables are articulated among themselves, forming the units we call feet, depending on how many they are and the position they occupy. It is a binary system: the limit of possible combinations is determined by our processing capacity. In languages whose phonological accent is based on stress, as is the case of Portuguese and Spanish, an equivalence is made while translating poetry.

See which feet usually occur, with their respective names and formalizations, in the decasyllable (ten-syllable verse), used in the sonnet "From the tower":

Iambic (I), whose formalization is - /, meaning a short or unstressed syllable, followed by a long or stressed syllable, as in the beginning of the first verse of the second stanza of "From the tower": "If not always implied, they are revealed."

Trocheu (T), whose formalization is/ -, meaning a long or stressed syllable, followed by a short or unstressed syllable, as in the last three words of the second verse: "*com poucos, porém doutos livros juntos*,", in English, "with few, although be learned readings jointly." In Brazilian Portuguese, the trochaic form predominates in nouns, adjectives, and verbs.

Pirríqueo (P), whose formalization is - -, meaning a short or unstressed syllable, followed by another short or unstressed syllable, as in the beginning of the third verse of the second stanza: "*e em silenciosos, músicos conjuntos*", in English, "in a counterpoint music silently."

Espondeu (E), whose formalization is//, meaning long or stressed syllable, followed by long or stressed syllable. The single example in "From the tower" is "Dom Ioseph."

Anapesto (A), whose formalization is - - /, meaning short or unstressed syllable, followed by another short or unstressed syllable, and another long or stressed syllable, like the first feet of the first line of the sonnet: "Retirado na paz destes desertos," in English, "In the peace of these deserts, my heart broken."

Dactylic (D), whose formalization is/- -, meaning a long or stressed syllable, followed by two short or unstressed syllables, as in the first feet of "*destes desertos*" in English, "peace of these deserts," from the first verse of the first stanza: "*Retirado na paz destes desertos,*" in English, "In the peace of these deserts, my heart broken."

8. Micro-analysis of "From the tower" translation

See **Figures 4** and **5**.

Da Torre (BP version)	From the Tower (English version)
Retirado na paz destes desertos, *com poucos, porém doutos livros juntos,* *vivo em conversação com os defuntos,* *e os mortos eu escuto, olhos despertos.*	In the peace of these deserts, my heart broken with few, although be learned readings jointly, in conversation with deads, I live coyly, I hear the deads, the eyes vigilant, open.
Se não sempre entendidos, descobertos, *emendam, ou fecundam meus assuntos;* *e em silenciosos, músicos conjuntos* *à vida em sonho falam sempre abertos.*	If not always discovered, ever woken, they amend my affairs eternally; in a counterpoint music silently to the dream of life, they speak awoken.
As grandes almas ceifa a morte ingente, e a *vingar as injúrias de outrora* os *doutos livros, Dom Ioseph, isentam.*	The noble souls that death from us excepts, the injuries avenging of back days in changeless flight the hour flees away: The learned books, Dom Joseph, them exempt.
Em fuga irrevogável foge a hora; mas sempre o melhor cálculo assenta no que a lição e estudos nos melhora.	But always the best calculus attempts in the lesson improving our ways.

Figure 4.
"Da Torre" (BP version) into English.

Desde la torre

Retirado en la paz de estos desiertos,
pero doctos libros juntos,
vivo en conversación con los difuntos,
y escucho con mis ojos a los muertos.

Si no siempre entendidos, siempre abiertos,
o enmiendan, o fecundan mis asuntos;
y en músicos callados contrapuntos
al sueño de la vida hablan despiertos.

Las grandes almas que la muerte ausenta,
de injurias de los años vengadora,
libra, ¡oh gran don Ioseph!, docta la emprenta.

En fuga irrevocable hoye la hora;
pero aquélla el mejor cálculo cuenta,
que en la lección y estudios nos mejora.

Figure 5.
Desde la torre (Quevedo's original).

9. Stylistic analysis and the search for aesthetic effects in translation

In the BP version first word, "*Retirado,*" we are faced at its polysemy, as it can have the meaning of someone who seeks peace in a retreat, as well as that of the banishment imposed on Quevedo: I lean toward the second interpretation, and I used the metaphor "my heart broken," image of suffering. In the second verse, the author alludes to the confiscation of his books, when arrested in 1639. The adjective "*doutos,*" in the second verse, translated into "learned" is repeated in the last verse of the first tercet: the same positions are maintained in the English translation.

In the first stanza, there were only difficulties in the fourth verse, as "mortos" does not rhyme with "deserts": I made syntactic transpositions, dragging "*os mortos*" to the beginning of the fourth verse (the same with "the deads" in the English translation) and drawing "*despertos*,'" which was at the end of the second stanza, to crown the end of the first one (the same with "vigilant, open'" in the English translation).

The effect of these changes was the topicalization of "the deads." It is worth emphasizing that this verse is one of the most beautiful, created by Quevedo. Not only does the author use synesthesia and an oxymoron, but, as it is so often the case in literature, he anticipates neuroscience scientific findings, by proving that, after the written word recognition by the reading neurons, its acoustic images are internally heard. A similar anticipation of the unconscious was recorded by Virgil [24], in the *Aeneid*, when he uses the expression "*alta in mente.*"

In the second stanza, I made some lexical changes in the last word of the third verse, due to the rhymes. I believe they did not cause major semantic violations. The first verse is bimembre, and this resource, very Petrarquean, was kept. The third verse, however, was problematic, as "counterpoints" does not end in "ly." To keep the meter, I reversed the order and translated "*e em silenciosos, músicos conjuntos*" into "'in a counterpoint music silently" and the oxymoron was kept. The word "counterpoint," in addition to be in Quevedo's original version, connoting antithesis, was one of the Baroque musical forms, cultivated by J. S. Bach. It was possible to keep in the translation the anastrophe in the fourth verse, another Baroque characteristic.

It was not possible to keep always the same lexicon to obtain the rhymes, because the word endings and its suffixes are phonologically distinct, in the two languages. Sometimes, I used a syntactic resource, transforming an adjective into an adverb, like "jointly" for "*juntos*" (first stanza, second verse) or "silently" for "*silenciosos*" (second stanza, third verse); other times I added words, reinforcing the original semantic field as the word "coyly" from "in conversation with deads, I live coyly," for "*vivo em conversação com os defuntos,*" (first stanza, third verse), or as the word "vigilant" from "I hear the deads, the eyes vigilant, open" for "*e os mortos eu escuto, olhos despertos*." The dearest topos to Spanish poetry, the brevity of life, opens the last tercet, closed in the last two lines by a correlation.

10. Computational semantics contributions

The methodology of the present research resulted primarily from manually constructed sources that could have benefited from computational supervised learning, since the manual specification and the automatic acquisition of knowledge are solidly interrelated; however, the automatic induction of semantic information is guided and dependent on manually specified information.

One of the issues most worked on by Computational Semantics is the automatic disambiguation of the meaning of words: "Senseval was the first open community-based evaluation exercise for Word Sense Disambiguation programs" [25], essential to literary translation, in particular, poetry. From the polysemic nature of words [26], ambiguity multiplies in the situational context of statements, in which the enunciator, although using the same *significant* [27], denotes a new meaning to the referent, thanks to time, which is never repeated, with all the repercussions on his/her prior knowledge.

Computational Semantics seeks to solve this paradox, making use of databases and computational statistics, through which it was possible to make available the contextual patterns where the most frequent meanings of a significant data occur [28].

The following questions remain for literary translation:

a. Are situational communicative contexts the same in spatially and temporally different cultures?

b. Are stylistic resources, such as meter, translatable, or are equivalences sought?

c. Although linguistic communication is based on the assumption that the receiver expects new information from the enunciator (given/new; topic/comment), why do we feel pleasure when rereading the same poem dozens of times, or why does the child protest when we change a word while retelling the same story?

11. Conclusions

To understand the strategies used for translating Quevedo's sonnet "From the tower," I framed them in a theoretical discussion: translation is a creative process. It allows interpreting, according to Gadamer [6] and Theodor [7], as the source text, capturing the linguistic meanings emerged from the author's intratextual and intertextual relationships with extra-linguistic information, coming from his experiences, the moment and the historical-cultural space, according to Coseriu [11]. For this reason, I began the chapter with a historical context and with an evocation of the remarkable episodes in Quevedo's life that served as a background for the topos in his work and, in particular, for the sonnet that is the main focus of the paper. Only in this way we will be able to capture the allusion to prison and isolation, in the first verse of "From the tower" and the solace he finds with the books company.

However, when it comes to poetry, in addition to the general translation characterization, this genre requires that aesthetic aspects be prioritized, as mentioned by Jakobson [15], Paes [4], Eco [14], Silvestri [16], and Schleiermacher [17], such as meter, rhythm, rhymes, alliteration, paronomasia, and figures that relate to the signifier. Some authors even deny the possibility of translating poetry like Humboldt [10]. For this reason, a great deal of space was devoted to the examination of the sonnet form and rhymes.

In this chapter, I presented the strategies I used in the translation of the sonnet "From the tower" to achieve aesthetic effects similar to those of the Brazilian Portuguese version, given the phonological and morphological differences between it and American English, so, it was quite difficult to benefit from Computational Semantics supervised learning,

Acknowledgements

The author published another article in Portuguese, based on the same theoretical references, dealing with the difficulties of poetic translation of Quevedo's sonnets, however, from the original Spanish into Brazilian Portuguese, entitled: "OS MORTOS EU ESCUTO, OLHOS DESPERTOS: HOMENAGEM A MOACYR SCLIAR" (THE DEAD I LISTEN WITH MY OPEN EYES: HOMAGE TO MOACYR SCLIAR), in the journal Lingüística Vol. 32-1, June 2016: 79-93 ISSN 2079-312X online ISSN 1132-0214 printed DOI: 10.5935/2079-312X.20160005.

Author details

Leonor Scliar-Cabral
Federal University of Santa Catarina (UFSC), Florianópolis, SC, Brazil

*Address all correspondence to: ppgl@contato.ufsc.br; eonorsc20@gmail.com

References

[1] Alonso D. Poesia espanhola, ensaio de métodos e limites estilísticos. 3rd ed. Brasília: National Institute Library (INL)/ Culture Education Ministry (MEC); 1960. p. 471. (pp. 395, 428/9, 376, 24)

[2] Scliar-Cabral L. Poesia espanhola do século de ouro. 1st ed. Letras Contemporâneas: Florianópolis; 1998. p. 104

[3] Grotius H, Dübner F, Cougny E. Epigrammatum Anthologia Pallatina. 1st ed. Paris: Editore Ambrosio Firmin; 1871. p. 150

[4] Paes JP. Gaveta de tradutor. 1st ed. São Paulo: Letras Contemporâneas; 1996. p. 160

[5] Arendt H. Walter Benjamin and the Task of the Translator. 1st ed. Schocken: Rio de Janeiro; 1968. p. 278. (pp. 79-80)

[6] Gadamer HG. Hermenêutica da obra de arte. 1st ed. São Paulo: Martins Fontes; 2010. p. 506. (pp. 241, 243, 237)

[7] Theodor E. Tradução – Ofício e Arte. 3rd ed. São Paulo: Cultrix; 1986. p. 150

[8] Sapir E. The emergence of the concept of personality in a study of cultures. In: Mandelbaum DG, editor. Selected Writings of Edward Sapir in Language Culture Personality. Berkeley, Los Angeles: University of California; 2020. pp. 194-208. DOI: 10.1525/9780520311893-010

[9] Whorf BL. In: Carroll JB, Levinson SC, Lee P, editors. Language, thought, and Reality: Selected Writings of Benjamin Lee Whorf. 2nd ed. Cambridge, MA: MIT Press; 2012. p. 457

[10] Humboldt W. Introdução ao Agamênon. In: Heidermann W, editor. Antologia Bilíngue. Clássicos da Teoria da Tradução. Alemão-Português. 2nd ed. Florianópolis: UFSC/Núcleo de Pesquisas em Literatura e Tradução; 2010. pp. 104-117

[11] Coseriu E. O falso e o verdadeiro na Teoria da Tradução. In: Heidermann W, editor. Antologia Bilíngue. Clássicos da Teoria da Tradução. Alemão-Português. 2nd ed. Florianópolis: UFSC/Núcleo de Pesquisas em Literatura e Tradução; 2010. pp. 250-289

[12] Paz O. Translation: Literature and letters. In: Schulte R, Biguenet J, editors. Theories of Translation: An Anthology of Essays from Dryden to Derrida. Chicago: The University of Chicago Press; 1992. pp. 152-162

[13] Manzoorul IS. Translatability and untranslatability. The case of Tagore's poems. Perspectives Studies in Translatology. 1995;**3**(1):55-65. DOI: 10.1080/0907676X.1995.9961248

[14] Eco U. Experiences in Translation. 1st ed. Toronto, Buffalo: University of Toronto Press; 2001. p. 135. (pp. 12, 13, 88, 14, 16, 21, 105)

[15] Jakobson R. Concluding statement: Linguistics and poetics. In: Sebeok T, editor. Style in Language. 2nd ed. Cambridge, MA: The MIT Press; 1964. pp. 350-377

[16] Silvestri L. Nugae – Teoría de la traducción. 1st ed. Buenos Aires: Simurg; 2003. p. 80. (pp. 16, 12)

[17] Schleiermacher E. Sobre os diferentes métodos de tradução. In: Heidermann W, editor. Antologia Bilíngue. Clássicos da Teoria da Tradução. Alemão-Português. 2nd ed. Florianópolis: UFSC/Núcleo

de Pesquisas em Literatura e Tradução; 2010. pp. 38-101

[18] Dollerup C. Poetry: Theory, practice and feed-back to theory. Tradterm. 1997;**4**(2):129-147. DOI: 10.11606/issn.2317-9511.tradterm.1997.49856

[19] Nietzsche. Sobre o problema da tradução. In: Heidermann W, editor. Antologia Bilíngue. Clássicos da Teoria da Tradução. Alemão-Português. 2nd ed. Florianópolis: UFSC/Núcleo de Pesquisas em Literatura e Tradução; 2010. pp. 195-199

[20] Borges JL. Obras completas (v. 2). 1st ed. São Paulo: Globo; 1999. p. 565. (p. 258)

[21] Goethe JW. Três trechos sobre tradução. In: Heidermann W, editor. Antologia Bilíngue. Clássicos da Teoria da Tradução. Alemão-Português. 2nd ed. Florianópolis: UFSC/Núcleo de Pesquisas em Literatura e Tradução; 2010. pp. 29-35

[22] Horta AB, Vianna FM, Rivera JJ. Poetas do século de ouro espanhol. Poetas del siglo de oro español. 1st ed. Brasília: Embajada de España; 2000. p. 343. (p. 57)

[23] Carvalho F. O sonho é nossa chama. 1st ed. João Pessoa: Expressão Gráfica Editora; 2010. p. 98

[24] Virgil L' Énéide. Nouvelle Édition Revue et Augmentée avec Introduction, Notes, Appendices et Index par Maurice Rat. 2nd ed. Paris: Garnier; 1955. p. 439. (p. 4, line 26)

[25] Kilgarriff A, Palmer M. Introduction, Special Issue on SENSEVAL: Evaluating Word Sense Disambiguation Programs. Computers and the Humanities. 2000;**34**:1-2

[26] Ravin Y, Leacock C. Polysemy: Theoretical and Computational Approaches. Oxford: OUP; 2000. p. 242

[27] De Saussure F. Cours de Linguistique Générale. Édition critique preparée par Mauro TD. Paris: Payot; 1972. p. 510

[28] Schütze R. Disambiguation and connectionism. In: Ravin Y, Leacock C, editors. Polysemy: Theoretical and Computational Approaches. Oxford: OUP; 2000. pp. 205-219

Section 4

Semantic Analysis in Computing

Chapter 6

Toward Lightweight Cryptography: A Survey

Mohammed Abujoodeh, Liana Tamimi and Radwan Tahboub

Abstract

The main problem in Internet of Things (IoT) security is how to find lightweight cryptosystems that are suitable for devices with limited capabilities. In this paper, a comprehensive literature survey that discusses the most prominent encryption algorithms used in device security in general and IoT devices in specific has been conducted. Many studies related to this field have been discussed to identify the most technical requirements of lightweight encryption systems to be compatible with variances in IoT devices. Also, we explored the results of security and performance of the AES algorithm in an attempt to study the algorithm performance for keeping an acceptable security level which makes it more adaptable to IoT devices as a lightweight encryption system.

Keywords: cyber, information security, IOT security, networks, cryptography, AES, lightweight cryptography

1. Introduction

An information system is a set of interconnected components that collect, process, store, and transfer information. These components include the physical and software components and the communication networks [1].

Networks enable communications between many devices by connecting them and enabling the most reliable possible connection. Moreover, networks are subject to many attacks due to users and their different directions. Here, the challenge lies in maintaining the security of these networks with their resources and data while maintaining high performance [1–3].

In its simplest sense, Internet of Things (IoT) is a system of various intelligent devices known in our daily lives. These things link and communicate between them and ensure data transfer between them independently via the network without human interaction, a self-control system [4–6]. Smart Cities played an essential role in highlighting IoT. Smart Cities express the concept that depends on the city's technology, as these cities are linked to each other electronically. Information is collected continuously from sensors, monitoring, and computers covering the whole city [5, 6]. "Thing" term can be a sensor network, as safe houses, or in general, any device that can take an IP address and can interact through a network [5].

Security plays an essential role in judging IoT applications strengths. Users wish to have secure IoT software hat is secure in all respects. IoT application's security includes a secure transfer of data, protection from eavesdropping, and unauthorized

access. The system security has become one of the essential critical requirements of the system's core functions [2, 3]. Furthermore, the security aims to achieve what is known as the Confidentiality, Integrity, and Availability (CIA) triad. Finally, one of the most critical security goals is to control access through the Authentication, Authorization, and Accounting (AAA) framework [3, 7].

IoT causes a massive increase in the volume of data. Securing such enormous amount of data requires special efforts. Several technologies may serve this purpose. But the devices used in the combination of smart cities and the IoT vary among themselves in capabilities. Moreover, most of these devices have limited specifications and restrictions [5, 6]. Hence the need to find new technologies that work on these limited capabilities and achieve an acceptable degree of security. Furthermore, since the capabilities are limited, these technologies should be lightweight and rely on simple operations without consuming energy, storage, and processing capacity.

The rest of the chapter organized as follows. We provide clarifications for some concepts related to this field considering cryptography and Lightweight Cryptography (LWC) areas in Section 2. Section 3 provides some researches related to LWC, and Advanced Standard Algorithm (AES). In Section 4, recommendations and findings are discussed. Finally, we conclude the paper and present a vision for future work in Section 5.

2. Background

This section introduces the concepts of IoT, Cryptography, and LWC in Subsections 2.1, 2.2, and 2.3 respectively.

2.1 Internet of thing

IoT today is a hot topic in research. The importance of IoT comes because of keeping pace with the variables of life that call us to exploit everything new in technology, such as computers, cars, TV, refrigerators, and washing machines [5, 6]. **Figure 1** shows IoT Reference Architecture. The figure shows that the IoT system consists of data collector's devices as a sensor used to get the data and data analyzer device like a mobile phone used for data processing to make a decision. These two subsystems communicate and transfer data via a network [5, 6, 8].

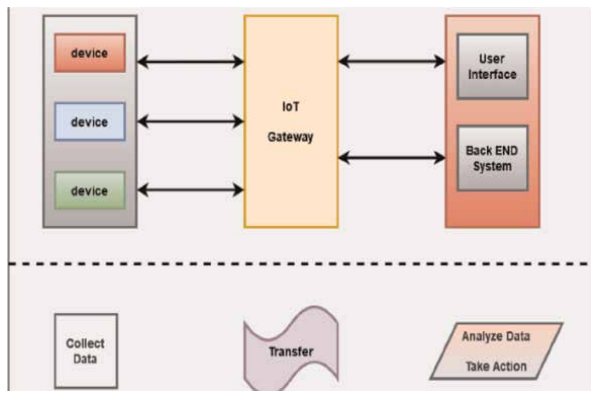

Figure 1.
IoT reference architecture [8].

IoT has dramatically helped to increase the efficiency of work and operations. It relies on a system of self-interaction that means reducing the waiting time for response. As a result, performance gains, and therefore the number of completed processes increases, giving users access to the best possible user services, enhancing the work's actual value [5, 8]. In general, IoT provides a wide range of benefits at the enterprise and individual levels.

Figure 2 presents the concept of IoT. There are many valuable and significant applications for IoT, such as Safe Houses, Health Care, and Farming systems.

Despite the significant benefits of IoT, the IoT suffers from a lack of standardization and is vulnerable to cyber-attacks, data theft, data fraud, botnet attacks, and physical compromises. The reason for this is that the IoT differs from traditional networks. There are two types of IoT devices: those rich in resources, like computers, and those with limited resources, like sensors. The real challenges are in the second type, which has low memory and computing power, short battery life, and Low bandwidth to connect [6, 8]. So, we should be careful about security and privacy [8]. Hence, the challenge is how to design an IoT system efficiently and securely.

2.2 Cryptography

Cryptography is a way to protect data and communications by ensuring that those not authorized to access sent data cannot read and process it [3, 9]. The goals of the encryption process revolve around guaranteeing each of the following [3, 9, 10]:

- *Confidentiality:* Using Encryption to protect data from unauthorized reading.
- *Data Integrity:* Ensures that the message remains the same as sent without changing it by using a unique message digest.
- *Non-Repudiation:* Ensures that the recipient does not deny the message's arrival by proving that the sender sent the message.
- *Authentication*: Proof of an entity identity, which confirms the user's right to access the system or data.
- *Access Control*: Ensures that access to the system or data is limited by preventing unauthorized access and checking their privileges.

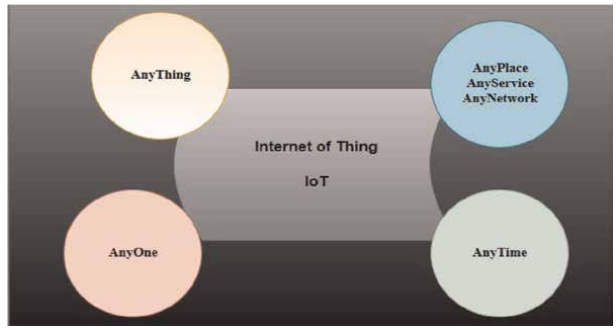

Figure 2.
Concept of IoT [4].

- Still, there are some essential terms related to security worlds, they include:

- *CIA Triad:* In addition to confidentiality and integrity, we still have the concept of availability, which ensures that authorized users can access what they want at any time. Therefore, the CIA triad tries to achieve the three goals emphasized [9].

- *AAA Framework* is responsible for enforcing policies and controlling access over resources. In addition to the authentication previously mentioned, it ensures that the security methods used in the network guarantee [7, 11]:

- *Authorization*: Not much different from access control. It works on the resources the user is allowed to access and use.

- *Accounting*: Directly, it can be defined as a complete monitoring process and writing down all the operations that the user performs to be used further in the accounting, analysis, and planning process.

Figure 3 summarizes CIA triad and AAA framework.

2.2.1 Cryptography algorithms

In cryptography science, encryption transforms original messages (Plain Text) to non-readable data (Cipher Text) using an encryption algorithm. This Cipher Text cannot give anyone any information about the Plain Text except those with the encryption key [9, 12–17]. Therefore, we can perform a simple encryption example by replacing every character in the plain text with its next character in aliphatic order.

P = "Thesis".

Alg.: substitution $P_i = P_{i+1}$.

C = "uiftjt".

There are two main types of encryptions: Asymmetric cipher, and Symmetric cipher, as shown in **Figure 4** [9, 12–17].

2.3 Asymmetric cipher

Asymmetric cipher is conjointly referred to as public-key cryptography. Associate cryptography technique uses a mix of public key and private key. The sender has the receiver's public key, whereas the private key is not known. The receiver ought to produce his try of the general public and private key, publish his public key while not

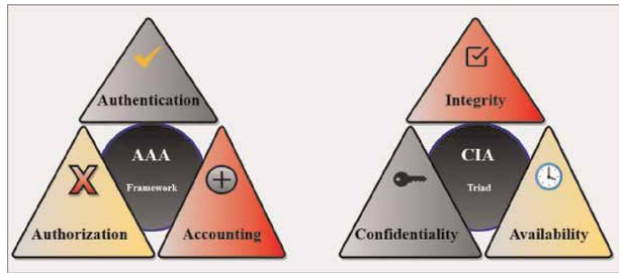

Figure 3.
CIA triad and AAA framework [10, 11].

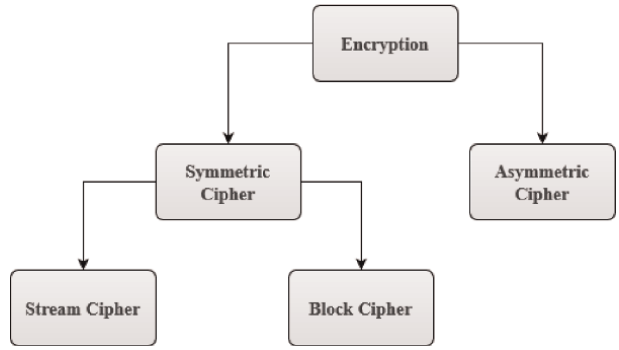

Figure 4.
Encryption models [9].

considering its security. The private key should be a procedure not possible to seek out through the general public key.

Uneven cryptography is employed in authentication and digital signatures. A signed message with the sender's private key proves the sender's identity, and anyone who has that sender's public key can verify it. Thus, the receiver may ensure that the message has not been changed or replaced by the other one that confirms the sender's identity [9, 15, 16].

Rivest–Shamir–Adleman (RSA) algorithm one of the most popular and widely used asymmetric encryption algorithms. It was developed in 1977 by Ron Rivest, Adi Shamir, Leonard Adleman and took its name from them. Besides Encryption, Digital signatures and key exchange are possible using RSA [17, 18].

RSA gained its strength by relying on parsing large integers in the formation of keys. First, two prime numbers are manipulated to create the user's public and private keys. Then, the message is encrypted using the recipient's public key and decrypted exclusively with the recipient's private key. **Figure 5** shows the RSA Process [17].

Although RSA is the most popular and secure asymmetric encryption algorithm in terms of key difficulty, it takes a long time to encrypt and decrypt. Besides, a security flaw appears that encrypting the same message again produces the same encrypted message [18].

ElGamal is an asymmetric cipher based on Diffie–Hellman key exchange. This algorithm gains its strength through the difficulty of finding discrete logarithms. For example, even though we know G^x and G^y, it is challenging to find G^{xy}. This algorithm consists of key generation, encryption, and decryption processes. **Figure 6** shows each of them [19–21].

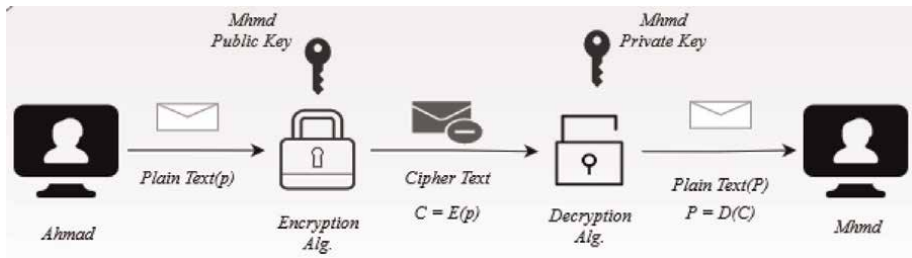

Figure 5.
RSA process [18].

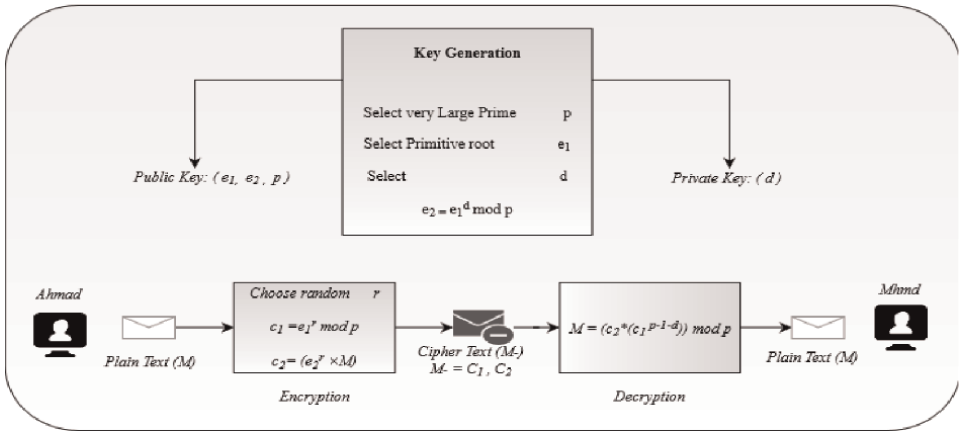

Figure 6.
ElGamal Alg. [20].

Elliptic Curve Cryptography (ECC) it uses the mathematics on elliptic curves. ECC is widely used due to its high security and small size. The difficulty in cracking the elliptic curves that underpin key strength has made ECC more secured and considered as the next generation of RSA [21, 22].

The main difference between ECC and RSA is the strength of the key. A 160-bit key in ECC is equivalent in power to a 1024-bit key in RSA. Considering that there is no linear relationship, doubling the size of the RSA key does not mean that we need to double the size of the RSA key. ECC is characterized by the speed of obtaining the keys and less memory to store them. On the other hand, a challenge for ECC is that it cannot be implemented as efficiently as RSA [22]. **Figure 7** presents the ECC.

Digital Signature Algorithm (DSA) is an algorithm that uses discrete logarithms and standard bases to introduce and validate the notion of a digital signature. Compared to RSA, DSA provides faster key generation.

As a result, it is slower in the encryption process, but it offers better results in the decryption process. DSA is mainly used to verify the sender's identity of a message since it bears his signature, which cannot be duplicated [23]. **Figure 8** presents the DSA mechanism.

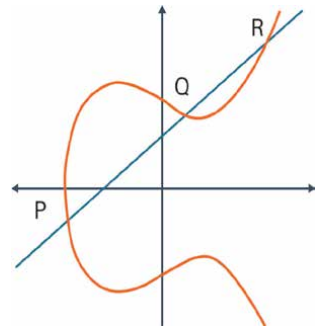

Figure 7.
ECC basics [22].

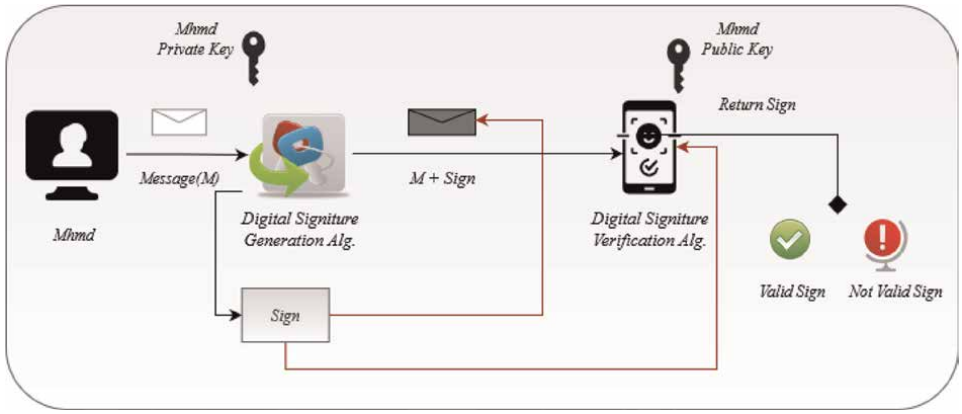

Figure 8.
DSA process [23].

2.4 Asymmetric cipher summary

This section discussed various Asymmetric Cipher algorithms such as RSA, ElGamal, DSA, and ECC. **Table 1** highlight the most comparison points between them. This type of algorithm offers high strength in terms of security, it requires a large amount of processing, which means low performance and draining resources. Therefore, based on the preceding, these algorithms are not compatible with the discrepancy in the capabilities of IoT devices and therefore cannot be used in building security systems in term of encryption. Hence, we find that symmetric encryption is more suitable for such systems. However, this does not detract from its value, as it cannot be dispensed with in verification, key exchange, and signature operations.

Cipher	Key size *(bits)*	Strength	Weakness
RSA	1024 2048 3072 4096	• Low computational time. • Fast.	• Use same module for multi users. • For small messages. • Not Scalable.
ElGamal	1024	• Fast. • Very efficient in hardware imp. • Solve discrete logarithm. • Good Scalability. • Low Power Consumption.	• Require Random Number Generator. • Ciphertext is very Large. • Slow in Signing.
DSA	512–1024 ***(multiple of 64)***	• Authentication. • Integrity. • Non-repudiation.	• Entropy. • Secrecy. • Uniqueness of random signature.
ECC	160 224 256	• Small Key size. • Low storage. • Low transmission time, and power consumption. • Very Fast.	• Ciphertext is large. • High Complexity

Table 1.
Asymmetric ciphers comparison.

2.5 Symmetric cipher

Each sender and receiver share the same secret key in this kind of Encryption. Hence, it uses within the encryption and decryption processes. However, symmetric Encryption has better speed but a lower security level than asymmetric [9, 12, 16, 24]. **Figure 9** shows the general structure of this encryption model. Symmetric ciphers can be used as a block cipher or stream cipher. We will discuss both types in detail in this section.

2.5.1 Stream cipher

In this type of encryption, the data are encrypted bit by bit. Because every encrypted bit is independent of other bits, diffusion and confusion properties are not achieved [9].

This encryption type mainly uses as simple as possible operators in this type of cipher. In most cases, it uses the XOR operation between the plaintext bits and the corresponding key bits. As a result, stream cipher throughput (speed of Encryption) is much *higher* than the block cipher but is considered less secure than Block Cipher [9, 12–16, 24].

Rivest Cipher 4 (RC4) is a stream cipher algorithm proposed by Ron Rivest in 1987. It later became a widely used algorithm from being a personal algorithm due to its speed and simplicity. RC4 has been frequently used to encrypt network traffic [25]. This algorithm uses byte-oriented operations with a variable key size. Simply put, RC4 relies on an XOR operation between each piece of plaintext with a small portion of the key to produce the ciphertext. And the decoding process is only a reflection of this process. However, with the development of computers, it became possible to break this algorithm easily. However, RC4 can be considered secure if the initial bytes of the key are ignored [25–27].

Salsa20 is a synchronous stream cipher suggested by Bernstein. The number 20 indicates the number of rounds, but this can be reduced to 12 or 8 as needed. *Salsa20* relies on simple operations such as rotation, addition, and XOR, making it a high-speed algorithm, which makes it secure against timing attacks [27, 28].

Sosemanuk is a synchronous stream cipher with variable key length. It has good properties of confusion and diffusion for a low cost. Furthermore, the Mux operation

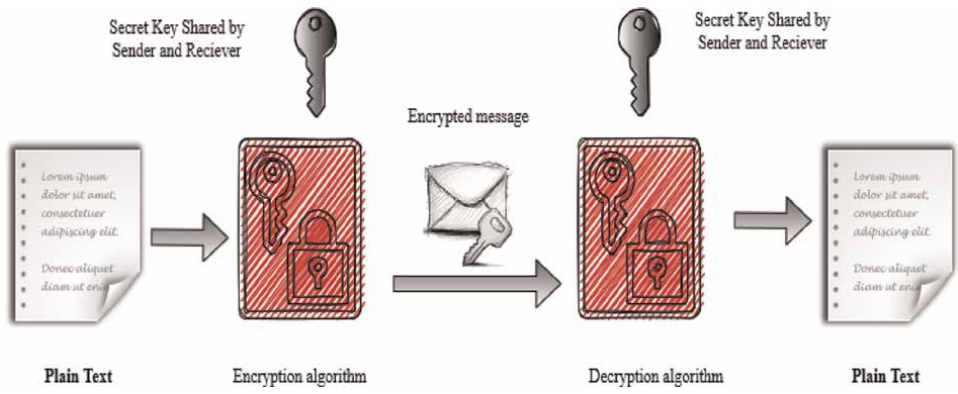

Figure 9.
Simple symmetric model [9].

is secure against algebraic and fast correlation attacks. Finally, Sosemanuk has good performance due to the internal static data [29].

Table 2 provides a brief comparison of these algorithms, following our discussion and our review of their definitions and specifications.

From this comparison, we note that the RC4 algorithm is optimal for use, as it is more robust and available in more than one version to suit the system in which it will be used. However, in light of the fact that stream ciphers offer high speed and low security and the requirement for keys to be the same size as plaintext, none of these algorithms are suitable for use as a foundation for building an IoT system.

2.5.2 Block cipher

In Block Cipher, the plaintext is divided into blocks based on encryption algorithm structure [12]. This type of Encryption has an execution time slower than the stream cipher. So, the encryption throughput of stream cipher is much higher than the block cipher [9, 23]. In contrast, a block cipher provides better security than the stream cipher against some well-known attacks. Moreover, the essential properties of the secure ciphertext, which are the confusion and the diffusion properties, are included inside block ciphering algorithms. Based on these facts, we can nominate one block cipher algorithm to build our algorithm for the IoT after reviewing it and choosing the most appropriate based on its specification and results.

Data Encryption Standard (DES) is a symmetric encryption algorithm that uses a seemingly 64-bit key, of which 56 bits are used as the practical key over 16 rounds of the 48-bit subkeys, to encrypt data of a fixed length of 64 bits. The apparent key's remaining 8 bits are utilized to verify for parity. In decryption, the same process is employed in reverse [30, 31]. **Figure 10** shows an example of DES encryption.

Even though this algorithm has been widely adopted due to its speed and ease of use, it suffers from a serious security weakness in reality. The use of DES with a short key makes it very fragile, especially using a brute force attack, which is easy to use in this case. In addition, there are many attacks, such as Davie's attack and offensive Linear and differential cryptanalysis, which are theoretical attacks [30, 31].

An improved version of the encryption algorithm has been created to solve the security issues with DES. This method is as simple as applying the DES algorithm precisely three times. We now have three keys, each of which is 56 bits long. As a result, the implementation technique differed in the keys utilized. There were several versions because the relationship of the three keys affects the extent of the algorithm's power in the previously described. Triple DES (3DES), which used three distinct keys with a total of 156 actual bits, was thought to be very powerful [28]. However, 3DES will not be used by the end of 2023 as we move to more secure generations for encryption [32].

Stream cipher	**Key size$_{bits}$**	**Data size $_{bits}$**	**Rounds**	**Speed$_{CPB}$**
RC4	1–2048	2046	1	7
SALSA20	128 256	512	20	3.91
SOSEMANUK	128–256	32	20–32	5.6

Table 2.
Stream cipher comparison.

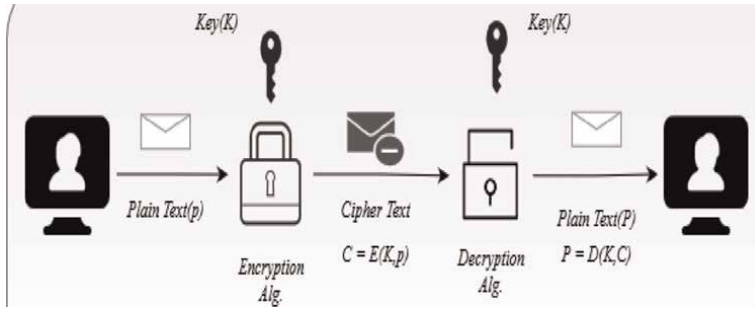

Figure 10.
DES algorithm [30].

Blowfish is a symmetric cipher technique that uses a 64-bit block and a variable-length encryption key as needed. In terms of speed, Blowfish is a good algorithm, but the amount of security it provides varies depending on the length of the key employed. As a result, even though no genuine threats have been detected, it has gotten less attention than other algorithms [32, 33].

AES is one of the most famous and prominent symmetric encryption algorithms that has been introduced to be a quantum leap in this field. AES has outstanding performance and an excellent security level compared to its peers.

AES deals with data blocks with a fixed size of 128 bits in length, in addition to providing flexibility in choosing the size of the key according to the required degree of security. From here, it appears that AES has three versions according to the size of the key, namely AES-128, AES-192, and AES-256 with 10, 12, and 14 rounds, respectively. Each process uses several operations to encrypt a data block [34–36]. **Figure 11** represents the flow of the AES algorithm.

The working mechanism of AES is based on the use of the design principle known as the permutation and substitution network, and this mechanism is represented by using the following arithmetic operations:

- *SubBytes:* Using a predefined look-up table known as Rijndael S-Box, each byte will be replaced with another one. Without any linear relation.

- *ShiftRows:* The row elements are swapped by shifting them cyclically to the left.

- *MixColoumns:* Using a linear transformation relationship, the change of all column elements is combined so that they affect each other to increase the level of difficulty through the propagation property.

- *AddRoundKey:* The data cells are combined with the subkey cells generated for this round using XOR operation.

The need for key expansion comes from the fact that each AES round needs a key of a specific length based on the criteria mentioned earlier. Therefore, when using AES-128, we need 11 keys depending on the number of rounds. Key derivation is done using the AES Key Schedule algorithm, which expands the key using a key schedule [34, 35].

AES distinguished itself from its peers in improving its performance for systems dealing with large amounts of data by integrating these steps and running them on a

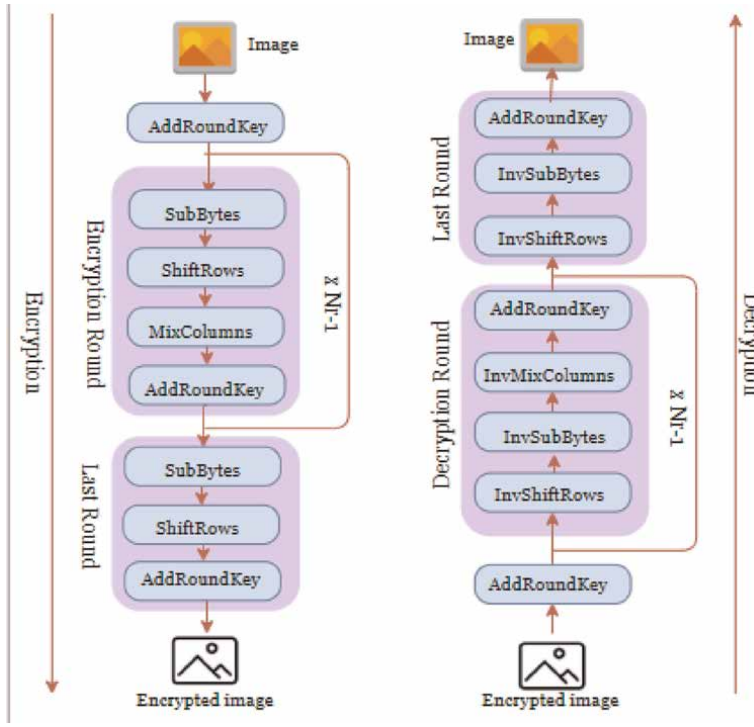

Figure 11.
AES algorithm [31].

byte-oriented approach. This approach only converts its arithmetic operations into a series of look-up tables [35].

3. Modes of operation

In Block Cipher, a fixed size block is handled at a time. Usually, the data size is much larger than the block size. Hence, the data is divided into a set of blocks. Each block is encrypted as one unit, the relationship, and dependency between encrypted blocks relaying on the encryption mode. Several modes have been developed to accommodate the variety of applications that will use Encryption. The process of selecting the required mode depends on many factors such as error propagation, the level of security, pre-processing, parallelization, and the speed of Encryption and decryption [12, 36]. These modes are as follows:

- *Electronic Code Book (ECB)*: It is an explicit and imperative coding process. It is considered the simplest since the text is split and each block is encrypted independently [36].

- *Cipher Block Chaining (CBC)*: This mode has constituted a development from the ECB. The block encryption process has become dependent on the result of the previous block encryption, which increased the data dependency on each other and made it possible non-deterministic. In this mode, the plaintext XOR-ed with the result of the previous block encryption before the encryption process [36].

- *Cipher Feedback (CFB)*: Looking at the CBC mode, this mode also relies on the result of the previous block as feedback to the present block and some other variables to increase the resistance to attacks [36].

- *Output Feedback (OFB)*: There is no difference between it and CFB except in some minor details that increased the resistance to bit errors and reduced the relationship of Encryption to plaintext [36].

- *Counter (CTR)*: It is a counter-based CFB. This mode is mainly based on maintaining the synchronization of the counter between the sender and receiver [36].

In general terms, without going into details of each mode. **Table 3** compares these modes.

After discussing the previous block cipher algorithms such as DES, 3DES, Blowfish, and AES, after reviewing the definition and specifications of each, **Table 4** provides a brief comparison of these algorithms.

From this comparison, we found that the stream has better performance and complexity, but it is not guaranteeing the diffusion, can be reversed easily, and it is providing less security. Because of that, we conclude that the block cipher is better solution since it provides more security in the case of text-based and image-based encryption.

3.1 Symmetric cipher summary

In this section, we summarize the symmetric cipher algorithms. **Table 5** compare stream and block cipher algorithms.

3.2 Cryptography summary

After discussing the cryptography algorithms and classifying them into Asymmetric and Symmetric, we reviewed their definition and specifications of each type. **Table 6** provides a brief comparison of these algorithms.

Mode		ECB	CBC	CFB	OFB	CTR
Padding Required		Yes	Yes	No	No	No
Error Propagation		No	All next block	Next block	No	No
Parallel	Enc	Yes	No	No	No	Yes
	Dec	Yes	Yes	Yes	No	Yes
Pre-Comp		No	No	No	Key	Yes
$Speed_{/5}$	Enc	5	2	1	3	4
	Dec	2	1	4	3	5
Security		Low	High	High	High	Medium
As Stream		No	No	Yes	Yes	Yes

Table 3.
Encryption modes comparison.

	DES	3DES		Blowfish	AES		
Key $_{bits}$	56	112	168	32-448	128	192	256
Block $_{bits}$	64	64		64	128		
Rounds	16	48		16	10	12	14
Security	Not	Secure		Moderate	Secure		
Speed	Slow	Very Slow		Fast	Fast		
Scalability	No	Yes		Yes	Yes		

Table 4.
Block cipher comparison.

	Stream	Block
Design	Complex	Simple
Data	Handle 1 byte at a time	Split data into a set of blocks
Number of bits	Depending on Block size	1 bit
Complexity	High	Low
Speed	Fast	Slow
Resources	Require more resources	Require fewer resources
Confusion and Diffusion	Confusion	Confusion and diffusion
Reversing	Simple	Hard
	Cannot take block cipher properties	It can be as a stream,

Table 5.
Stream vs. block cipher.

3.3 Lightweight cryptography

NIST defined LWC as a cryptosystem whose features have been optimized to meet the requirements of devices of varying specifications, especially resource-constrained devices [37]. From this definition, we conclude that all cryptography terms can be LWC if it is possible to legalize its need for resources to ensure the desired effect. Thus, asymmetric Encryption is an exception due to its complexity and demand for high resources. On the other hand, symmetric Encryption can be used in these systems if it is properly exploited.

Depending on the critical challenges mentioned before, we found that the LWC algorithm should use little memory and power and provide good performance while maintaining the required level of security [38]. Therefore, the factors of LWC requirements can be explained as follows [39]:

- *Key Size:* Longer Key size is better for security, but it requires more complexity and power.

- *Block Size:* smaller block size is more familiar with IoT since the big block size requires more CPU, memory, and power.

	Asymmetric	Symmetric
Keys	Two keys; one for Encryption and the other for decryption	Single Key for Encryption and decryption
Key Exchange	Not a problem	Big Problem
Relation between number of keys and receivers	# of Keys = (# of receivers) *2	# Of Keys = # of receivers
Cipher Size	Same or Larger than plain text size	Same or Smaller than plain text size
Speed	Slow	Fast
Data Size	Used for small data	Used for Large data
Provide	Confidentiality, authenticity, and non-repudiation	Confidentiality
Key Encryption and recourses utilization	High	Low
Examples	RSA, ElGamal, ECC, and DSA	RC4, Salsa20, Sosemanuk, DES, 3DES, Blowfish, and AES

Table 6.
Asymmetric cipher comparison.

- *The number of rounds:* Fewer rounds are better since the rounds require more computation and resources.

- *Structure:* The structure here is the way of managing the trade-off between all previous factors to find the optimal combination to ensure an acceptable level of performance and security.

Many LWC algorithms provide different performance and security strengths. And after studying many related studies, we find that there have been some trends in relying on stream cipher due to its high efficiency in terms of performance. Still, most of the algorithms were based on block cipher since it offers better security but with a significant performance improvement [38]. We highlight some of these LWC algorithms in the following sections depending on its base as a stream or block.

3.4 Stream LWC

This section presents some LWC algorithms based on stream cipher methodologies.

A4 is a very efficient lightweight stream cipher that uses LFSR and FCSR. The key feature of A4 is the ease of implementation and high security. In addition, A4 has proven itself in resistance to brute-force and algebraic attacks [39].

New Lightweight Stream Cipher (NLSC) is a chaos-based algorithm that uses an 80-bit secret key, two Nonlinear Feedback Shift Registers (NFCR), and three multiplexers. NFCR has good security, making it resistant to statistical attacks and providing good performance [38, 39].

3.5 Block LWC

This section presents some LWC algorithms based on block cipher methodologies.

PRESENT is an LCW algorithm that relies on Substitution-Permutation Network (SPN). It was suitable for limited hardware as it uses an 80-bit key. However, it was noted that it takes 32 rounds to encrypt 64 bits. Another version uses a 128-bit key, but it requires more computations [38].

GIFT is an enhanced PRESENT version; it uses a lighter S-Box with minimal rounds and a faster key scheduling algorithm. These properties enable it to provide more throughput. It is also available in more than one version depending on the required throughput. These versions are; GIFT-64 and GIFT-128. With a 64-bit block size that requires 28 rounds and a 128-bit block size that requires 40 rounds, respectively [38].

KATAN is an algorithm that outperforms PRESENT by saving 48% of the power. KATAN uses an 80-bit key and handles different text sizes 32, 48, and 64 bits. However, its downside is that it requires 254 rounds to complete this process [38].

The National Security Agency developed Simon as an improved algorithm that uses rounds cycles but uses a lot of arithmetic operations. It offers many different key sizes as 64-bit, 72-bit, 96-bit, 128-bit, 144-bit, 192-bit, and 256-bit that handle 32-bit, 48-bit, 64-bit, 96-bit, and 128-bit block size through 32, 36, 42, 44, 52, 54, 68, 69, and 72 rounds. While SPECK is the same as SIMON, it supports exact block sizes and keys, but 22, 23, 26-29, and 32-34 rounds [38].

RECTANGLE is a very LWC algorithm, which is different from PRESENT. It Relies on lighter SPN with 25 rounds. This reduced algorithm significantly speeded up the execution based on Bit-slice, as it relies on parallel swapping and replacement [40].

SIT is an algorithm that combines Feistel and SP network and takes five rounds to handle 64 bits of text with 64 bits of text as a key. It mainly consists of two parts: the first is for key expansion, and the second is for the encryption section. Key expansion

	Key $_{bits}$	**Block $_{bits}$**	**Rounds**	**Sec.**	**Characteristics**
A4	128	—	—	—	Secure High performance
NLSC	80	—	—	—	Secure Good performance.
PRESENT	80 128	64	32	80%	Low memory Suitable for small data
GIFT	128	64 128	28 40	85%	Simple Fast Key Scheduling. High throughput
KATAN	80	32 48 64	256	—	Inefficient. Low throughput Energy consuming
SIMON	64–256	32–128	32–72	67%	High performance Easy and Flexible
SPECK			22–34	58%	As SIMON but optimized for software
RECTANGLE	80 128	64	25	60%	Fast Hardware Friendly
SIT	64	64	5	—	Fast Key Scheduling High throughput Need low energy

Table 7.
LWC summary.

relies on simple operations such as concatenation, shifting, addition, and XOR. As a result, this algorithm achieves high throughput and low power consumption [38].

3.6 LWC summary

After discussing various LWC algorithms such as LSC, A4, NLSC, PRESENT, GIFT, KATAN, SIMON, and SPECK, RECTANGLE, and SIT. **Table 7** highlight the most comparison points between them.

4. Literature review

This section discusses the most recent related research. After studying these researches, we categorized them into two groups. The first group, including [40–50], reviews LWC and defines its essential requirements. The second group discusses AES versions that are proposed to be compatible with LWC requirements [51–60].

4.1 Lightweight cryptography related works

This section summarizes some researches that introduce the concept of LWC in terms of terminology, requirements, and how to implement them in line with the available capabilities.

Manifavas et al. [40] discussed lightweight encryption algorithms, focusing on streaming encryption, which provides high performance with simple operations, making it suitable for the capabilities of IoT devices, especially when the text length is unknown or continuous. The results showed the superiority of symmetric encryption in performance. Still, most of the streaming algorithms were not secure, as after analyzing 31 algorithms, it was found that only 6 were secure.

Buchanan et al. [41] emphasized the IoT's security and privacy challenges. Also, the researchers review the trends of designing lightweight algorithms after explaining alternatives to traditional cryptography methods that fit the composition of the IoT. Finally, after reviewing the challenges in terms of physical and software implementation, the study recommended that when developing LWC solutions, the following should be noted:

- Resorting to small blocks and a short key constitutes a security weakness and leads to faster wear of CBC mode.
- The number of operations is directly proportional to the size of the inputs; in lightweight symmetric cipher almost twice.
- The algorithm architecture must be adapted to new applications and better integrate with existing protocols.

Based on the previously mentioned recommendations, the following are the methods presented by this study that can be included when designing a lightweight security system for the IoT [41]:

- ***Hashing***: is a mathematical algorithm that assigns data of arbitrary size (often called "message") to a fixed-size bit matrix (the "message summary"). It is a one-way

function that is practically useless to reverse or reverse the account. Ideally, the only way to find a message that produces a particular hash is to forcibly search for potential inputs to see if they have a match or use a rainbow table of identical hash.

- ***Streaming***: it is a symmetric key cipher in which the plaintext is combined with a string of pseudorandom, keystream characters. In-stream cipher, each plaintext character is encoded with its corresponding character from the stream key to giving the ciphertext characters. An alternative name is state encoding, stream cipher, where the encoding of each character depends on the current state. The character is usually a bit and operation (XOR) or exclusive-or in practice.

- ***Block***: It's an encryption method that applies a deterministic algorithm along with a symmetric key to encrypt a block of text, rather than encrypting one bit at a time as in stream ciphers. For example, a typical block cipher, AES, encrypts 128-bit blocks with a key of a predefined length: 128, 192, or 256 bits. Block ciphers are Pseudo-Random-Permutation (PRP) families that operate on a fixed-size block of bits. PRPs are functions that are computationally indistinguishable from random permutations and, therefore, are considered reliable until their unreliability is proven.

Sehrawat et al. [42] presented a detailed comparison between several algorithms compatible with the IoT and after conducting cryptanalysis attacks. This study also showed that block ciphers had attracted the attention of many researchers as a basis for developing LWC algorithms. Finally, this study also recommended the requirements for the future of LWC algorithms.

Dutta et al. [43], reviewed the encryption solutions that can be used in the IoT by comparing some LWC that can fit with the nature of IoT devices. Researchers believe that symmetric encryption is the closest to suit the heart of the IoT. They also found that the modified AES algorithm provides a suitable security solution to the restrictions imposed by the capabilities of IoT devices after studying many algorithms like *DES, 3DES, Blowfish*, etc.

After choosing AES as a standard and reliable algorithm and achieving the desired goal, the researchers analyzed the performance of a set of versions of the algorithm implemented in previous studies by sorting them into two parts as follow [43]:

- **Recent Research Work on AES for IoT**: Many implementations achieved good results in high productivity, low energy, and minimal costs.

- **Recent Research Work on AES for IoT Focusing MixColumns and S-box**: The researchers focused on this aspect of the hardware implementation. Delay and reducing the area are the main goals of algorithm development, so the main challenge that exists to date is to improve Mix-column round and S-box operations. There are many implementations as the Serpent Algorithm that were previously developed to meet the challenges mentioned [43]. With these designs, we can provide good results to achieve these goals.

The researchers also presented a study of attacks on AES that should be monitored and found solutions such as Differential Fault Analysis Attacks and wireless interceptive side-channel attack techniques. These attacks can be resisted through the use of dummy keys and XOR operations [43].

Rajesh et al. [44] presented the Novel Tiny Symmetric encryption Algorithm (NTSA), which provides better confusion for each round which leads to better security level. The comparison centered with the TEA algorithm is considered one of the most attractive algorithms because of its ease of implementation and less memory usage. Its main problem is to use the same key for all rounds, which reduces the level of security and its poor performance. The results show that NTSA outperforms many other security algorithms and achieves better performance, making it more suitable for IoT and embedded devices.

Gunathilake et al. [45] discussed the future applications of LWC, how to implement it, and the challenges it faces. The study also touched on the existing LWC algorithms previously mentioned in our research and confirmed the effectiveness of the modified AES algorithm in this field.

Usman et al. [46] reviews the light encryption algorithms that fit the nature of the IoT after identifying the obstacles to using traditional algorithms, such as the low power capacity of the devices. Researchers believe that the security of big data flowing through the IoT is the main problem, as this weakness may overwhelm the advantages of IoT applications. Therefore, considering the capabilities of these devices represented in the low capacities, it was necessary to think of new methods that require simpler arithmetic operations and less memory while providing an acceptable degree of security. In addition to what has been mentioned, these methods must consider the diversity of devices, their different capabilities, and the protocols used to have the ability to integrate and adapt to this diversity. And now we still have the issue of privacy, as the IoT, with the vast amounts of data circulating, must provide the user with the possibility of appropriate control over his data [46]. The researchers considered that symmetric encryption is best suited for the IoT because asymmetric encryption requires higher capabilities. And the following are some of the symmetric encryption algorithms that have been reviewed [46].

Abutair et al. [47] believe that despite their importance, smart cities still face the challenge of balancing the quality of service and maintaining the privacy and security of information. This study summarized the difficulty of achieving this balance as follows:

- Design limitations and limited capabilities make this environment an easy target for hacking.
- The truth of the data may be injected to cause damage, leading to great disasters.
- Difficulty of building a standardized system due to different manufacturers.

The researchers studied many lightweight algorithms used in the IoT. Based on this study, an infrastructure has been proposed that provides a specific degree of privacy and security for the IoT. This study concluded that some modern algorithms such as *CLEFIA* and *TRIVIUM* achieved terrible results compared to the old algorithms, especially *TRIVIUM*, which gave disastrous results [48]. The study explains the structure of smart cities. Without going into details here, the aspect that concerns us is the necessity of providing IoT devices with algorithms that meet the guarantee of authentication, integration, and confidentiality to protect the network from threats. Such as *Corrupted Data, Replay Attacks, IP Spoofing, Identity Usurpation, DoS/DDoS Attacks, and, Data Leakage* [47]. This study presented a new design that depends on the capabilities of the device that will be added. Based on these capabilities, the

appropriate lightweight algorithm is selected for it. The mechanism of this design can be summarized as follows:

- *Input:* Device specifications
- *Knowledge Base:* minimal requirements for each lightweight algorithm.
- *Output:* The appropriate algorithm for this device.

After testing many algorithms by changing some factors, the researchers found that the algorithm closest to adapting to the majority of IoT devices is the *AES* algorithm, with the need to reduce its resources [47].

Ramadan et al. [48] introduced a LWC algorithm called LBC-IoT that handles 32-bit blocks with a key of up to 80 bits. This algorithm is based mainly on the Feistel structure, along with simple operations such as XOR that do not consume power and 4-bit S-boxes. The results indicate the strength of this algorithm against attacks in addition to its acceptable performance, and it is considered a promising algorithm for implementation on small and very restricted devices.

Periasamy et al. [49] proposed a lightweight block cipher mechanism that works on 8-bit processing, as their study indicates that this algorithm is superior to its counterparts. According to the researchers, this algorithm derives its strength from the strength of the encryption in the compensation boxes. In terms of performance, the design of the compensation boxes played marginally using the Multi sequence Linear Feedback Shift Register and reliance on simple operations such as XOR, shifting, and registers to reduce space required and optimization in power consumption and speed.

Thabit et al. [50], researchers introduced a New LWC Algorithm (NLCA) to secure cloud computing applications. This algorithm uses a 16-byte key based on Feistel and substitution permutation. This algorithm succeeded in achieving confusion and diffusion by introducing some logical operations into the algorithm's formula, such as Shifting, Swapping, and XOR. One of the advantages of this algorithm is the flexibility, such as AES, where the number of rounds and the length of the key are variable according to the application's needs. The results also indicate that this algorithm provides a good level of security and performance, which makes it suitable for these applications.

In this section, we discuss many LWC related researches. **Table 8**, focus on the key points that have been discussed in IoT cryptography related works and summarize them.

4.2 AES related works

In this section, we summarize some researches that present some AES-based system, discuss these systems and highlight the differences in these AES versions to reach the best possible ways to improve the performance and strength of this algorithm more.

Javed et al. [51], presented a new design for the AES algorithm to make it suitable for mobile devices and speed it up despite the limitations of the hardware specifications. After reviewing the mechanism of the standard AES algorithm, the researchers discuss the improvement that was made to AES implementation and the motives that were relied upon in this optimization as follows:

Study	Key points
2015 [40]	• Discuss many LWC algorithm. • It shows that the symmetric encryption is very good in performance, but most symmetric algorithms are not secure.
2018 [41]	• Discuss IoT security challenges. • Recommendations to be followed when developing LWC.
2018 [42]	• Compare many security algorithms that compatible with IoT. • It shows that the block cipher algorithms are more suitable to be used. • It also recommended the requirements for the future of LWC algorithms.
2019 [43]	• Discuss some encryption techniques that can be used in IoT. • Discuss AES algorithm.
2019 [44]	• Propose NTSA which provide good security level.
2019 [45]	• Discuss the future of LWC and its challenges.
2020 [46]	• Discuss some LWC algorithms. • It recommended the adoption of symmetric encryption because asymmetric encryption requires powerful resources, and this is what IoT devices lack.
2020 [47]	• Discuss many LWC algorithms. • The study found that modern algorithms did not meet the requirements due to poor results. • The study recommended the use of AES due to its strength, provided that it is configured to improve performance.
2021 [48]	• Propose LBC-IoT which provide very good performance with low power consumption.
2021 [49]	• Propose a new lightweight block cipher which provide a good security and performance.
2021 [50]	• Propose NLCA which used to secure the cloud, and it provide very good security with accepted performance.

Table 8.
LWC related works summary.

- This optimization used a 10-byte look-up table for round constant and two 256-bytes look-up tables for S-box and InvS-box. The constant round means that the three rightmost bytes are always 0. Thus, XOR performed only on the leftmost byte of the word. The round constant differs from one round to the other.

- ***In MixColumns***, the multiplication with 02 can be performed by a left shift and bitwise XOR with 1b.

- ***In ShiftRow***, Using the row index as a specific number (i), each row is rotated to the left by i. This implies that the first row will not be rotated.

- ***In RoundKey***, a rounded key is added to the State matrix by a simple bitwise XOR operation: a sum in the field GF (2^8). Each round key is obtained from the key schedule.

- There are two ways to implement Key scheduling: (1) key unrolling (2) On the fly key generation. This study implements key unrolling because that *On the fly key*

generation approach is costly in clock rounds and need 16 bytes of additional memory to store the last round keys for the decryption [51].

The results of this study showed that the performance of the proposed method gives better results, as it provides 3 times better encryption speed and is about 20 times better in round keys calculations. This design outperformed its predecessor by 20 times while reading data from the hard disk and encrypting it if the data was greater or equal to 1 MB [53].

Abhijith et al. [52], presented an improved model for implementing the AES algorithm by slicing and integrating the internal processes of the algorithm. This new version used Block-Ram and 10 levels of pipelines to improve efficiency and productivity. The results indicate that this enhanced version significantly enhances performance and the possibility of integrating it with other systems.

Bui et al. [53] worked on finding an improved version of AES in several ways. First, reduce the combinational logic and number of records by organizing the data path. Second, the clock gateway strategy, key expansion, and minimization of data activities contributed to reducing the algorithm's energy use. Here are the modifications that have been implemented to achieve the above improvements:

- By using the Low Power S-Box, power consumption is reduced.
- Logic relationships were reduced by manipulating data by columns after eliminating ShiftRow.
- Using a special mechanism to load data and encryption keys limits the number of records.
- Finally, the clock gate scheme worked in reducing energy consumption.

These modifications were additions that can be used without modifying the algorithm. As for the fundamental alterations in the algorithm, they were represented as follows [53]:

- ***Thirty-Two-Bit Datapath Optimizations:*** The Advanced Low Power Encryption Standard (AES) can be used in smaller applications such as small-scale IoT devices. Proposed 32-bit AES data paths to meet low energy consumption and small space requirements. We only use the 32-bit data path in MixColumns.
- ***Substitution Box:*** The S-box takes several input bits (m) and converts them into a certain number of output bits (n), where n is not necessarily equal to m. m $\times$ n S-box can be performed as a search table with 2 million words each n bits. Fixed tables are usually used.
- ***Key Expansion Optimizations:*** The expansion was implemented in VHDL, resulting in ascending design and test methodology. This choice also ensures that the code can be transferred to different vendors' devices. The code and simulation were manufactured using Altera MAX + PLUS II version 7.21 Student Edition. The FPGA family was selected for execution from Altera Flex 10 K. It's part of an 8-bit execution with a 128-bit block and a 128-bit key. Because the goal

of improvement is to reduce consumption, to suit it for mobile applications, the structure is directed to minimize space.

The results show that the proposed version offers the same PRESENT algorithm in energy use. Also, the proposed system is resistant to the attack of power correlation analysis with less than 20,000 traces, which seeks to expose the data path. Also, the data path in case of parallelism provides it with more robustness. Finally, this design uses different key sizes, which contributes to providing various levels of security as needed [55].

Mamoun et al. [54] provided a comprehensive explanation of the AES algorithm. The study presented a new model for the AES algorithm to enhance its security level by adding an XOR operation to an extra byte of s-box and using an additional random key. The results indicate that this modification contributed to improving the level of AES security variably due to the randomness of the added key. The results also showed that this modification improved confusion and increased time security.

Umer et al. [55] tested AES using different techniques depending on the resources of the target devices, the results were characterized by varying in nature according to the techniques used. Among these techniques were used; Parallelization and storage of s-box and key expansion, as it has been noted that the introduction of such technologies helps in optimizing the exploitation of resources to provide better results.

Daoud et al. [56], the researchers present an optimization of the AES algorithm using Vivado High-Level Synthesis (HLS), and their results show significant progress in increasing the throughput of the proposed algorithm, which was implemented on the FPGA only using flip flops and look-up tables. Since optimizing commands in Hardware Description Languages (HDL) is not easy and time-consuming, HLS improves the algorithm with less effort. HLS is an automated process that deals with high-level programming languages such as C that is used to ease the struggles that HDL requires in the development process, debugging, and provide flexibility in meeting system requirements. HLS tool synthesized compiled core AES functions in an RTL block, and sub-functions were divided into sub-blocks at higher system levels. Below is a review of the improvements that this study made to the AES algorithm [58]:

- ***Key Expansion-based Implementation:*** key expansion process combined with the encryption process so that the two processes will run simultaneously during each round.

- ***SW-based Implementation:*** Key extension process is performed before the encryption process to obtain 11 different 128-bit keys based on AES-128 design.

- ***High Throughput-based Optimization:*** The algorithm has made some special optimizations to increase the encryption throughput.

The main objective of this study was to achieve the maximum throughput in encryption. The process that most positively affected the results is integrating key expansion with encryption. By comparing the effects of frequency, productivity, and area utilization, it appears to us that the proposed design in this study has outperformed the previous strategies [56].

Proceeding from the fact that the AES algorithm is considered the best secure algorithm currently available and can be adapted to IoT devices. Rokan et al. [57] provided an integrated security system for the IoT called *Modified Lightweight AES*

(***MLAES***) that includes two integrated systems; The first one is a *Secure Encryption* based on a lightweight version AES integrated with Chaos Maps. The second is a *Secure Authentication* using a chaotic hash function based on SHA3-256-bits. The following is a review of the three main phases of this system:

- ***Lightweight Modified AES:*** The goal of mitigating and optimizing AES is to reduce computational complexity, execution time and reduce required iterations and memory used. One of the most important modifications is the use of 4 chaos keys, which increases the randomness of results, which means enhancing system security. The first modification in the algorithm uses *shifting operations*, *data blocks*, and *logical functions*. MLAES uses two sub-boxes, each dealing with 64 bits of data. The second modification is to make the number of times of rounds and ShiftRow are executed dynamically based on a dynamic number. This number is generated depending on some chaos keys that change with each iteration. Finally, the last modification is to eliminate the MixColumns operation due it its complexity and high execution time by replacing it with some XOR operations, SHA3-128, and shift operations.

- ***Modified Sub-Bytes(S-Box):*** The s-box represents one of the complex operations in MLAES and is directly related to the degree of security of the design; S-Box takes 128 bits of data and divides it into 16-bit blocks. Every 64 bits of data is sent to a sub-S-box, where the system contains 2 S-Boxes. The S-Box shifted after each iteration using K to change its values.

- ***The Proposed IoT Security System:*** As mentioned earlier, besides the MLAES encryption process described in the previous points, the proposed system includes a hashing stage using SHA3-256.

The study results indicate that despite the modification to AES, the level of security remained strong, in addition to the significant improvement in its performance and the specifications required for its operation. Perhaps the most prominent result was that this system passed the NIST tests, which means that the system is resistant to linear differential attacks and brute force attacks [57].

Farooq et al. [58], given the discrepancy between the capabilities of IoT devices, explored five implementations of the AES algorithm. These applications use modifications and improvements to the AES algorithm. The applications indicate the disparity in the results, as each of these applications fits a specific category of IoT devices. Therefore, the study recommended moving away from comprehensiveness and not limiting encryption to one algorithm for all devices, but instead relying on the device's capabilities to choose the optimal AES version for use.

Nagalakshmi et al. [59], given the discrepancy between the capabilities of IoT devices, presented some strategies for implementing AES with a set of other systems to suit these devices of varying powers, and the study also touched on the use of LFSR. The results indicate a security improvement, the ability to check signatures, and random checks without significantly affecting performance.

Salim et al. [60] presented the development of an AES algorithm called multi-key AES. The name came concerning the fact that this proposal uses the AES algorithm but uses several keys as the secret key is used to configure a variable number of keys using ECC. The study specialized in implementing this algorithm in the IoT, provided that it is used on devices capable of running this algorithm. The results indicated that this

Study	Key points
2010 [40]	• Use new look-up tables for S-box and InvS-box. • Optimize MixColumn, ShiftRow, and RoundKey. • These optimizations enhance AES performance.
2017 [41]	• Reduce combinational logic and number of records. • Use Low- Power S-Box, and clock gate scheme. • Eliminate ShiftRow. • These optimizations enhance AES performance.
2019 [42]	• Using Vivado HLS which enhance the throughput of AES.
2019 [43]	• Propose MLAES which provide a secure encryption AES-based algorithm and secure authentication used chaotic hash function. • Enhance AES by use 4 chaos keys to improve security. And use two sub-boxes.
2014 [44]	• Improve AES by use slicing and integrating processes, block-RAM, and 10 level pipelines. • These modifications enhance the performance of AES.
2017 [45]	• Provide a comprehensive study of AES. • Enhance AES by adding an XOR operation to an extra byte which enhance the security of AES. • This enhancement improves the time security and confusion.
2021 [46]	• Use AES but with several keys based on ECC. • This optimization improves AES security, but it did not enhance its performance.
2020 [47]	• Using 5 modified AES models and studying their results, the study recommended not relying on the same algorithm on all devices, but rather choosing the appropriate algorithm for the capabilities of each device.
2020 [48]	• Modifying AES by using LFSR. • This modification enhances the security of AES, but it did not improve its performance.
2017 [49]	• By testing AES, this study shows that the improvement in AES performance can be done by parallelization storage of S-Box, and key expansion.

Table 9.
AES related works summary.

modification did not affect the algorithm's performance, but it contributed to improving its security.

In this section, we discuss many AES-based related researches. **Table 9**, present the summary of some researches that worked on modifying AES to adapt it with IoT.

5. Evaluation

This section presents the ways of evaluating algorithms and a brief discussion of this study.

5.1 Evaluation

The evaluation process should address performance evaluation and security evaluation to ensure the power of the algorithm. To evaluate performance, we will initially need to calculate the following:

- ***Execution Time***: is one of the essential parameters for evaluation performance. It measures the time needed to encrypt and decrypt a specific data size [61, 62].

- ***Throughput***: it reflects how much data can be processed during a time. It presents the average of data in kb divided by the average Encryption or Decryption time.

As for security, we will initially need to account for:

- ***Key Time Security:*** the time to attack the algorithm using brute force. Which is related to key size [9, 63].

- ***Histogram:*** study the uniformity of data distribution [9, 61].

- ***Confusion:*** study the relationship between ***ciphertext*** and ***key***; this relation should be robust. In simple words, the changing of 1-bit in secret key should lead to a significant change in ciphertext [62].

- ***Diffusion:*** study the relationship between ***ciphertext*** and ***plain text***; in simple words, changing 1-bit in plain text should affect the ciphertext highly [62].

- ***NIST Tests:*** These tests attempt to test the randomness of binary sequences produced by an algorithm. These tests focus on different types of non-randomness that could exist in a binary sequence. It was released by the National Institution of Standards and Technology (NIST) as a suite for testing PRNGs that contains 188 tests, including 15 main tests [9, 63].

		ECB	CBC	CFB	OFB	CTR
Key Time Security				2^{128}		
Enc. Time (s)	1	1.136	1.132	1.224	1.374	1.355
	2	228.899	243.123	246.811	242.619	241.472
Dec. Time (s)	1	1.32	1.41	1.14	1.05	1.12
	2	299.26	306.90	242.76	248.48	244.63
Enc. Throughput	1	1278.24	1063.87	1098.20	1192.91	1132.80
	2	1134.55	1037.38	1028.12	1051.15	1046.48
Dec. Throughput	1	987.11	866.45	1091.38	1169.09	1113.62
	2	885.73	826.89	1038.35	1047.64	1040.17
Histogram		8194.44	240.91	251.80	274.40	257.44
Confusion (%)		50.08	50.04	49.93	50.11	49.95
Diffusion (%)		0.08	50.16	49.82	0.01	0.01
NIST		13	15	15	15	15

[1] *155 KB Data.*
[2] *31 MB Data.*

Table 10.
AES evaluation results.

5.2 Summary

Based on all that was mentioned previously, studies have confirmed that stream cipher provides better performance than block cipher. Still, a block cipher is superior to a stream cipher in terms of security especially when we looking to better confidentiality. Some previous studies also indicated that lightweight stream cipher did not succeed much on the security front. From here, we can be sure that the basis in our research should be based on a block cipher with its security strength while trying to improve it in the level of performance [64].

We believe that using a recognized and standard algorithm to improve it would be better at the current stage. Most previous studies confirmed that the choice fell on AES due to its superiority. In appendix A, we review the summary of the results of the AES algorithm test in terms of performance and security to be a starting point for improvement [64]. These results are shown in **Table 10**.

These results showed that AES provide an acceptable degree of security according to this evaluation criteria, such as Key security, Histogram, NIST, Confusion, and Diffusion. But to prove that, we will use more security tests in future work such as Mapping, Correlation, Unified averaged changed intensity, and Number of Changing pixel Rate. On other hand, the result of performance testing can be improved by changing or replacing some core functions on AES.

6. Conclusion

In this chapter, a detailed study of computer security has been conducted. After clarifying different kinds of cryptography, LWC has been addressed, considering its basics and requirements. Some of the presented algorithms highlight the essential needs for LWC algorithms and the importance of making them compatible with the resources of IoT devices. This study also discussed the latest studies related to each of Lightweight Cryptography, Lightweight AES-based algorithms, and the most prominent evaluation criteria used to judge the suitability of an algorithm. Finally, this study presented the results of testing the AES algorithm according to the specified criteria. We believe that these results constitute a starting point for future work as promising results in the field of LWC algorithms and their suitability to the resources of IoT devices.

DOI: http://dx.doi.org/10.5772/intechopen.109334

Author details

Mohammed Abujoodeh*, Liana Tamimi and Radwan Tahboub
College of IT and Computer Engineering, Palestine Polytechnic University, Hebron, Palestine

*Address all correspondence to: 131089@ppu.edu.ps

IntechOpen

References

[1] Alfred Y. Network Security. Malaysia: Asia Pacific University; 2019. pp. 5-11. DOI: 10.13140/RG.2.2.19900.59526

[2] Manuj A. Network Security with pfSense: Architect, Deploy, and Operate Enterprise-Grade Firewalls. 1st ed. Birmingham: PACKT Publishing; 2018

[3] William S. Cryptography and Network Security: Principles and Practice. 8th ed. London: Pearson; 2017

[4] Mohammed Z, Ahmed E. Internet of things applications, challenges and related future technologies. Journal of World Scientific News. 2017;**67**(2):126-148

[5] Available from: https://internetofthingsagenda.techtarget.com/Ultimate-IoT-implementation-guide-for-businesses [Accessed: February 15, 2022]

[6] Mista S, Roy C, Mukherjee A. Introduction to Industrial Internet of Things and Industry 4.0. 1st ed. Florida: CRC Press; 2021

[7] Qabajeh L. A more secure and scalable routing protocol for mobile ad hoc networks. Security and Communication Networks. 2013;**6**:286-308

[8] Makhdoom I, Abolhasan M, Ni W. Blockchain for IoT: The challenges and a way forward. International Council for Evangelical Theological Education. 2018: 594-605

[9] Salhab O, Jweihan N, AbuJoodeh M, Abutaha M, Farajallah M. Survey paper: Pseudo-random number generators and security tests. Journal of Theoretical and Applied Information Technology. 2018;**96**: 1951-1970

[10] Abutaha M, Farajallah M, Tahboub RM. Survey paper: Cryptography is the science of information security. International Journal of Computer Science and Security. 2011;**5**:475

[11] Available from: https://geek-university.com/ccna-security/aaa-explained [Accessed: February 15, 2022]

[12] Elminaam DA, Abdual-Kader HM, Hadhoud MM. Evaluating the performance of symmetric encryption algorithms. IJ Network Security. 2010; **10**:216-222

[13] Guanrong C, Ybin M, Charles C. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons & Fractals. 2004;**21**:749-761

[14] Zhu S. Algorithm design of secure data message transmission based on Openssl and Vpn. Journal of Theoretical & Applied Information Technology. 2013;**48**:562-569

[15] Bellare M, Rogaway P. Optimal asymmetric encryption. In: Workshop on the Theory and Application of Cryptographic Techniques. Berlin, Heidelberg: Springer; 1994. pp. 92-111

[16] Simmons G. Symmetric and asymmetric Encryption. ACM Computing Surveys (CSUR). 1979;**1**:305-330

[17] Shamir A, Adleman L, Rivest R. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM. 1978;**21**: 120-126

[18] Alsadeh A, Karakra A. A-RSA: Augmented RSA. 40th conf of SAI Computing. 2016:1016-1023

[19] Gamal T. A public-key cryptosystem and a signature scheme based on discrete

logarithms. IEEE Transactions on Information Theory. 1985;**31**:469-472

[20] Available from: https://www.quest10.com/p/33937/el-gamal-cryptography-algorithm-1/ [Accessed: February 15, 2022]

[21] Maurer U, Wolf S. The Diffie–Hellman protocol. Designs Codes and Cryptography. 2000;**19**:147-171

[22] Mihailescu M, Nita S. "Elliptic-curve cryptography", elliptic-curve cryptography. In: Pro Cryptography and Cryptanalysis. Berkeley, CA: Apress; 2021. DOI: 10.1007/978-1-4842-6367-9_1

[23] Al-Absi M, Abdullaev A, Absi A, Sain M, Lee H. Cryptography survey of DSS and DSA. In: Advances in Materials and Manufacturing Engineering, Lecture Notes in Mechanical Engineering. Singapore: Springer; 2020. pp. 661-669

[24] Agrawal M, Mishra P. A comparative survey on symmetric key encryption techniques. International Journal on Computer Science and Engineering. 2012;**4**:877-882

[25] Stallings W. The RC4 stream encryption algorithm. In: Cryptography and Network Security, Prentice Hall, 2005

[26] Mister S, Tavares S. Cryptanalysis of RC4-like ciphers. Selected Areas in Cryptography. 1998;**1556**:131-143

[27] Robshaw M, Billet O. New Stream Cipher Designs: The eSTREAM Finalists. New York: Springer; 2008

[28] Bernstein D. The Salsa20 family of stream ciphers. In: Robshaw M, Billet O, editors. New Stream Cipher Designs. Lecture Notes in Computer Science. Vol. 4986. Berlin, Heidelberg: Springer; 2008

[29] Berbain C, Billet O, Canteaut A, Courtois N, Gilbert H, Henri L, et al. SOSEMANUK: A Fast Software-Oriented Stream Cipher. New York: Springer; 2008. pp. 98-118

[30] Coppersmith D, Holloway C, Matyas S, Zunic N. The data encryption standard. Information Security Technical Report. 1997;**2**:22-24

[31] Daemen J, Joan, Rijmen V. The data encryption standard. In: The Design of Rijndael. Springer; 2002. pp. 81-87

[32] Available from: https://www.cryptomathic.com/news-events/blog/3des-is-officially-being-retired. [Accessed: February 15, 2022]

[33] Bhat P, Deepthi. Comparison of MD5 and blowfish algorithm. International Journal of Innovative Research in Science Engineering and Technology. 2016;**5**:506-511

[34] Kalaiselvi RC, Vennila M. An analysis of AES, RSA, and blowfish - A review. The International Journal of Analytical and Experimental Modal Analysis. 2020;**XII**:568-588

[35] Blumenthal U, Fabio M, Keith M. The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-Based Security Model. Vol. No. RFC 3826. USA: Bell Labs; 2004

[36] Dworkin M. Recommendation for Block Cipher Modes of Operation: Methods and Techniques. SP: NIST; 2001. pp. 800-38A

[37] Cruz-Cunha M, Portela I. Handbook of Research on Digital Crime, Cyberspace Security, and Information Assurance. 1st ed. Pennsylvania: IGI Global; 2014

[38] Thakor V, Razzaque MA, Khandaker M. Lightweight cryptography

algorithms for resource-constrained IoT devices: A review, comparison, and research opportunities. IEEE Access. 2021;**9**:28177-28193

[39] Muhammad R, Quazi M, Rafiqul I. Current lightweight cryptography protocols in Smart City IoT networks: A survey. ArXiv. 2010;**00852**: 2020

[40] Manifavas H, Hatzivasilis G, Fysarakis K, Papaefstathiou J. A survey of lightweight stream ciphers for embedded systems. Security and Communication Networks. 2015;**9**: 1226-1246

[41] Buchanan W, Li S, Asif R. Lightweight cryptography methods. Journal of Cyber Security Technology. 2018;**1**:187-201

[42] Sehrawat D, Gill N. Lightweight block ciphers for IoT based applications: A review. International Journal of Applied Engineering Research. 2018;**13**: 2258-2270

[43] Dutta I, Ghosh B, Bayoumi N. Lightweight Cryptography for Internet of Insecure Things: A Survey. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference; 2019. pp. 0475-0481

[44] Rajesh S, Paul V, Menon V, Khosravi MM. A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded IoT devices. Symmetry. 2019;**11**(2):293

[45] Gunathilake N, Buchanan W, Asif R, Rameez. Next Generation Lightweight Cryptography for Smart IoT Devices: Implementation, Challenges and Applications. IEEE 5th World Forum on Internet of Things (WF-IoT); 2019. pp. 707-710

[46] Usman M. Lightweight encryption for the low powered IOT devices. arXiv. 2020;**2012**:00193

[47] Abu-tair M, Djahel S, Perry P, Scotney B, Zia U, Carracedo J, et al. Towards secure and privacy-preserving IoT enabled smart home: Architecture and experimental study. Sensor. 2020; **20**:6131

[48] Ramadan R, Aboshosha B, Yadav K, Alseadoon I, Kashout M, Elhoseny M. LBC-IoT: Lightweight block cipher for IoT constraint devices. CMC-computers Materials Continua. 2021;**67**:3563-3579

[49] Prakasam P, Madheswaran M, Sujith KP, Shohel S. An enhanced energy efficient lightweight cryptography method for various IoT devices. ICT Express. 2021;**7**:487-492

[50] Thabit F, Alhomdy S, Al-ahdal A, Abdulrazzaq, Jagtap P. A new lightweight cryptographic algorithm for enhancing data security In cloud computing. Global Transitions. 2021;**2**: 91-99

[51] Javed A. Fast Implementation of AES on Mobile Devices. Proc. 8th Int. Netw. Conf.; 2010. pp. 133-142

[52] Abhijith P, Goswami M, Tadi S, Pandey K. Optimized architecture for AES. Cryptology ePrint Archive: Report. 2014;**1**:540

[53] Bui D, Puschini D, Bacles-Min S, Beigné E, Tran X-T. AES Datapath optimization strategies for low-power low-energy multisecurity-level internet-of-thing applications. IEEE Transactions on Very Large-Scale Integration (VLSI) Systems. 2017;**25**:3281-3290

[54] Mamun A, Rahman S, Shaon T, Hossain A. Security analysis of AES and enhancing its security by modifying

S-box with an additional byte. International Journal of Computer Networks & Communications. 2017;**9**: 69-88

[55] Farooq U, Aslam F. Comparative analysis of different AES implementation techniques for efficient resource usage and better performance of an FPGA. Journal of King Saud University - Computer and Information Sciences. 2017;**29**:295-302

[56] Daoud L, Hussein F, Rafla N. Optimization of Advanced Encryption Standard (AES) Using Vivado High Level Synthesis (HLS). Proceedings of 34th International Conference on Computers and Their Applications. 2019: 36-44

[57] Naif R, Abdul-Majeed GH, Farhan AK. Secure IOT system based on chaos-modified lightweight AES. 2019 International Conference on Advanced Science and Engineering (ICOASE). 2019:1-6

[58] Farooq U, Mushtaq M, Bhatti M. Efficient AES implementation for better resource usage and performance of IoTs. In: CYBER 2020 - 5th International Conference on Cyber-Technologies and Cyber-Systems. France: Nice; 2020

[59] Nagalakshmi E, Mohan V, Kumar D. AES datapath optimization strategies for low-power low-energy multi security-level internet-of-thing applications. International Journal of Advanced Research in Science, Engineering and Technology. 2020;**2**:347-355

[60] Salim K, Alalak S, Jawad M. Improved image security in internet of thing (IoT) using multiple key AES. Baghdad Science Journal. 2021;**18**: 417-429

[61] Jagtap S, Thabit F, Alhomdy S. Security analysis and performance evaluation of a new lightweight cryptographic algorithm for cloud computing environment. Global Transitions Proceedings. 2021;**2**:100-110

[62] Coskun B, Memon N. Confusion/ diffusion capabilities of some robust hash functions. 40th Conference Information Sciences and Systems. 2006; **CISS'6**:1188-1193

[63] Farajallah M, Abutaha M, Abujoodeh M, Salhab O, Jweihan N. Pseudo-random number generator based on look-up table and chaotic maps. Journal of Theoretical and Applied Information Technology. 2020;**98**:3130

[64] Abujoodeh M, Tamimi L, Tahboub R. Exploring and Adapting AES Algorithm for Optimal Use as a Lightweight IoT Crypto Algorithm, [master thesis]. Palestine: Palestine Polytechnic University; 2022. Available from: https://scholar.ppu.edu/handle/123456789/8635

Chapter 7

Perspective Chapter: Computation of Wind Turbine Power Generation, Anomaly Detection and Predictive Maintenance

Cristian Bosch and Ricardo Simon-Carbajo

Abstract

Early power loss detection in wind turbines is a key for the wind energy industry to avoid elevated maintenance costs and reduce the uncertainty regarding generated power estimations. Location, especially of those wind farms isolated offshore, causes the strategy of scheduled-only maintenance inefficient and very costly, additionally presenting a typically long downtime after a breakdown. These problems point to the creation of predictive solutions to anticipate the maintenance procedure, preparing the necessary parts and avoiding the possibility of destructive failures. Predicting failures in structures of such complexity requires modeling their multiple components individually in addition to the whole system. For this purpose, physics-based and data-driven models are used, which have proven themselves in this context. Machine learning has proven to be a valuable resource for solving a variety of problems in this industry. Thus, we will propose data-driven Deep Learning methods to compute the Power output of wind turbines with respect to all the mechanical and electrical features by using two types of Deep Neural Networks: a simpler combination of linear layers and a Long-Short Term Memory Neural Network. Then, with the use of a one-dimensional Convolutional Neural Network we will predict the time to failure of the system.

Keywords: wind-turbine, predictive maintenance, computational semantics, deep learning, LSTM, time series, regression, classification, CNN

1. Introduction

As per WindEurope [1], wind energy represents the second biggest provider of energy in the European Union (EU), accounting for a 18.8% of the capacity, behind gas. Ireland, for instance, represents the 3.5% of the EU's combined capacity, and wind energy covers a 28% of the country's energy demand. In this specific circumstance, it is interesting to point that the maintenance cost of a wind turbine can go from a 16% to a 30% [2, 3] of the Levelized Cost Of Electricity (LCOE). Wind power technologies, on the rise despite being mature, would more greatly consolidate by

adopting solid solutions to the maintenance problem. Subsequently, the predictive approach for wind turbines and farms has become a priority, introducing techniques aiming for downtime reduction and wind turbine lifespan extension.

While physics-based modeling systems exists, our purpose is to approach the problem through the application of Deep Learning (DL) algorithms on the data collected by the Condition Monitoring System (CMS) and the SCADA control system from several wind turbines. For this work, we have obtained data from an onshore Siemens SWT-2.3-101 wind turbine. The target is, based on historical data, to predict the anomalous behavior of the system and a fault with enough anticipation.

This chapter will explain the phases of data preparation, acknowledging there is a time series behavior and the application of Deep Learning (DL) solutions. Two Deep Neural Network architectures designed with the purpose of data-driven prediction of the SCADA measured "Power" output have been trained in a semi-supervised paradigm, assuming the training samples belonged to what could be considered the normal state of the wind turbine. We will address whether there is a significant improvement if the regression on the power output is done with a basic Artificial Neural Network (ANN) mainly built on linear layers or using a memory cell, as it is the case in Long-Short Term Memory ANNs (LSTMs). For the latter, our choice will be applying directly of Bidirectional LSTMs (BiLSTMs) as they are expected to offer a performance superior to that of the base architecture [4]. Then, the anomalies in the prediction of validation and test datasets will be found using several methods that assume that, ideally, the prediction deviations from real data will follow a normal distribution (which will be the driver of the hyperparameter optimization). These methods include the standard deviation, a Monte Carlo dropout and a few-shot dropout using less predictions and fitting a t-Student distribution on them, all three at 95% CL. The anomalies will be statistically analyzed afterwards aiming for testing hypothesis about their relationship to the Time-to-Fault, which is highly complex to generalize in the case of these datasets with errors of different origin and short periods between them (the maximum constant performance period being of 11 days). The statistical significance of the anomaly appearance will be a motivator for building a one-dimensional Convolutional Neural Network (1-dim CNN) to predict downtime with an appropriate time-to-fault. In this classification task, we will consider our "Prefault" label for a processed sample as another hyperparameter and draw critical conclusions of its possible values. For a previous reference on our work in this dataset, we studied data augmentation through optimal transport dataset aggregation in [5].

The book chapter has the following structure: The subsequent subsection displays the state of the art with regards to wind turbine fault forecasting and general anomaly detection using several Artificial Intelligence methods. After that, we explain the methodology of our approach to solving this problem, analyzing the data with regards to discriminating them semantically as a preparation for the computation of the power output and the detection of anomalous behavior. Then, we will explain the methodology for our data-driven regression of the power output with the use of two different deep neural networks and the possible the extraction of the anomalies in the predicted data. After showing statistical interest in these anomalies, we will proceed to show our strategy to classify data samples with a 1D-CNN and present how an interesting choice of metrics and hyperparameter space can help solve this problem. We finalize the book chapter by sharing the conclusions of our research.

1.1 State of the art

Wind power generators are composed of different rotating components that undergo an extensive overall performance during its lifetime. Condition Monitoring Systems (CMSs) are common within the modern industry and are a set of sensors that screen the state of the turbine's one-of-a-kind components in real time. The topic has extensively been reviewed in [6], where the advantages of Fault Detection Systems are outlined, ranging from cost reduction to the improvement of the Capacity Factor, since the ability to forecast and anomaly can optimize the moment when the maintenance is applied, avoiding any stops at high energy output periods. A CMS can collect data from a wide range of sensors focused on vibration, component temperatures, oil levels and electricity voltages and currents. Fault forecasting can combine this strategy with monitoring processes affecting the wind turbine, such as: crack detection, strain, thermographic analysis, electrical conditions, signal and performance monitoring and acoustic analysis.

There are methods inside the literature which put their attention on modeling the activity of the wind turbine parts by means of their physical behavior [7] and enhance this model with CMSs data from the wind turbine to create an approach favored by this hybridization. However, the challenge of our research is to consider only the data aggregated from the CMS to model the behavior of the wind turbines and make the model useful to predict periods of downtime and determine if it must be in a general way or with specific information about the upcoming fault.

Concerning the employment of these data using ML algorithms, the problem of defect detection can be resolved with two different strategies: i) Modeling the normal performance and detecting anomalies as they arise and ii) Evaluating data from time spanning before faults to anticipate defective behavior. We will face the problem by one or a combination of both these approaches.

Regarding anomaly detection, there is an advantage in modeling the normal output periods of a wind turbine, which is the use of most of the data collected by the CMS, as the datasets present a big imbalance with normal regime being the most populated class. These are known as semi-supervised models, since faults and time ranges of data close to faults are removed purposely before training the algorithms. A representative solution of this strategy are autoencoders. Autoencoders are Deep Neural Networks (NNs) with a symmetrical architecture with encoder layers that arrive to a bottleneck that stores the encoded representation of the data. These data will be decoded afterwards, with an output that preserves the important features. Autoencoders have become a reference in early fault detection and have been proven capable of discriminating the parts originating the failure [8]. There are other possibilities for early anomaly detection present in the literature: We can find NN architectures trained only with normal regime data that predict the power output expected at the next time iteration which, once compared against the actual generated output of the turbine, can determine if its behavior is unexpected. The parts responsible for such faulty behavior can be traced through a Principal Component Analysis (PCA) analysis [9]. We will partially follow this technique, modeling the power output through our dataset and then deviating from that work in the way anomalies are studied. Classification methods that constrain normal behavior periods to be modeled only if far prior or far posterior from a fault have been proven to be a competent way of discriminating the SCADA delivered features worth of consideration [10].

Moving to putting the focus on the historical faults of a device, the range of techniques is diverse too. The literature contains classifiers relying on supervised

training, which uses datasets with every sample labeled. One example that differs from the analysis of SCADA data turns to visually inspect the turbines with drones and then trains Convolutional Neural Networks (CNNs) to detect usual vortex generator damage indicators such as erosion or missing gear teeth [11]. These datasets, while based on image collection instead of SCADA, feature high data imbalance too, which requires compensation from software, providing complex architectures for the CNNs proposed for the task [12]. With respect to using turbine sensor data in fully supervised training scenarios, multiclass classification with Support Vector Machines (SVM) has been undertaken with simulated turbine data, showing success in discriminating faults according to their nature [13]. SVMs gained early popularity in predictive maintenance field, though the main models currently employed are decision trees and gradient boosting. This is shown in the benchmarking of Random Forest and XGBoost classifiers present in [14]. Signal analysis has been experimented on too [15], where interference in the currents of the Double-Fed Inductor Generator (DFIG), originated by the vibrations of the faulty gearbox are studied through autoencoders and NN classifiers for anomaly detection.

Focusing more on the work related to anomaly detection [9], we can also find several applications of ANNs for different time series problems. We will follow the strategy of dividing the dataset in four pieces [16], emphasizing that training must be done with what is considered normal regime data [17], which we will apply to both our ANN based on linear layers and an BiLSTM. These methods rely on the deviations from predictions to the real output to fall in a normal distribution, so as to have a reliable way of computing confidence intervals for the anomalies to be spotted [18, 19]. The confidence intervals associated to the prediction will be computed making use of the Monte Carlo dropout, activating the dropout layers in the evaluation time for the computation of a big number of predictions, a method use by Uber [20]. In order to prove the viability of less resource-consuming, a small number of dropout activated predictions will be computed and then fitted to a t-Student distribution [21]. We will try to predict anomalous behavior using the global standard deviation of predictions in the normal validation dataset too, for a more complete analysis.

Wind turbine data feature engineering can be a complicated task. The idea of using CNNs, famous for extracting the interesting features of images, in the field of fault prediction, has been explored in the literature too, as a feature information extraction tool to be combined with LSTMs [22] and as an independent predictor for software bugs [23]. Another model that is a combination of adaptive feature engineering and a CNN for fault forecasting can be found in [24]. Techniques aiming for automated feature engineering and modeling are a greatly explored topic in the field of data science, as they ease building models when domain expertise is not available. Among these, a couple of very relevant toolkits are AutoML [25, 26] and the H20.ai package [27, 28].

2. Methodology

2.1 Data

Our data originates from a Siemens SWT-2.3-101 turbine. Samples were collected every ten minutes and the dataset spans for a period of nearly four years. From the features included in the dataset, we choose those that refer to weather conditions (wind speed, temperature, etc.), have mechanical origin (gear bearing temperatures,

blade angles or pressure, etc.) or electrical measurements (different voltages and currents) to train our models.

The dataset is cleaned and labeled according to the status flag associated to each sample and assumptions with regards to posterior status flags. The computation of the regression requires a semi-supervised approach: After we make an initial split in training, validation and test sets, being the test a fourth part of the original data and the validation part a fourth portion of the remainder, we purge the training data from what we cannot confirm as well-behaved samples, and then we make the same in the validation dataset, thus creating one with only well-behaved data and a full validation dataset. All these portions are chronologically ordered, since we are dealing with a time series. This could have implications when training the BiLSTM, as there would be cuts due to the preprocessing purges. We will later assess if this has caused relevant effects on the models trained.

Since we are using a real industrial dataset that has not been curated for research purposes, there is no labeling beyond the indication of a fault happening (status flag). The definition of the normal or good behavior we seek to isolate is not clear (normal is not used as belonging to a Gaussian distribution here). As a way to deal with it, we define normal data based on the number of days before a fault is registered. This number of days will be considered a hyperparameter and thus the labeling process is included as part of the optimization to avoid putting a human bias greater than using full days as a time reference, which is a flexible enough decision. Trimming the data this way ensures that the power regression is computed with samples that are not semantically different, which will help achieving a well-behaved prediction distribution. A scaler is fit in what is considered normal data and the transformation is applied on the whole dataset, as the normal data is confirmed not to contain any outliers on features that would provoke a loss of information after scaling with the use of extreme points. The posterior classification task will have a similar labeling strategy, where a logistic regression will be performed for classifying each sample as either "Normal" (or 0) or "Prefault" (or 1). These prefault periods will be determined by a hyperparameter as well, which will then be determined by the best model during optimization.

As we are aiming for a full data-driven regression, after the selection of the features, there will be no dimensionality reduction as we want to include every detail in the prediction of the power. This contrasts with theoretical approaches that would try to reproduce the power curve, which is considered a relationship mostly exclusive between "Wind Speed" and "Power".

Moving to statistical features of the data, as it being normally distributed, it is a fact that the wind turbine works within a regime that makes the power curve a relation but not a function (**Figure 1**), since each value of the domain can correspond to several values of the codomain. This has, as we will expand later, significance when the intent is to have a regression where its errors are normally distributed. Our intent is that the deviations of the real data from the prediction (on the well-behaved regime used for training and validation of the regression) are normally distributed which, due to the power output being multivalued, is not trivial (see **Figure 2**).

Considering that the dataset only once shows a period of 11 days going without any downtime and that we will assume that, for the convenience of scheduling the maintenance, at least 24 hours of anticipation are needed, we will set the days of well-behaved regime hyperparameter between 2 and 6 days before a fault, so the optimization will decide the best possible outcome without slicing dramatically the amount of training samples.

The metrics of the regression will be measured in the well-behaved split of the validation dataset, and it will be the full validation dataset the one used to compute statistics referring to the appearance of anomalies in the prediction of the power

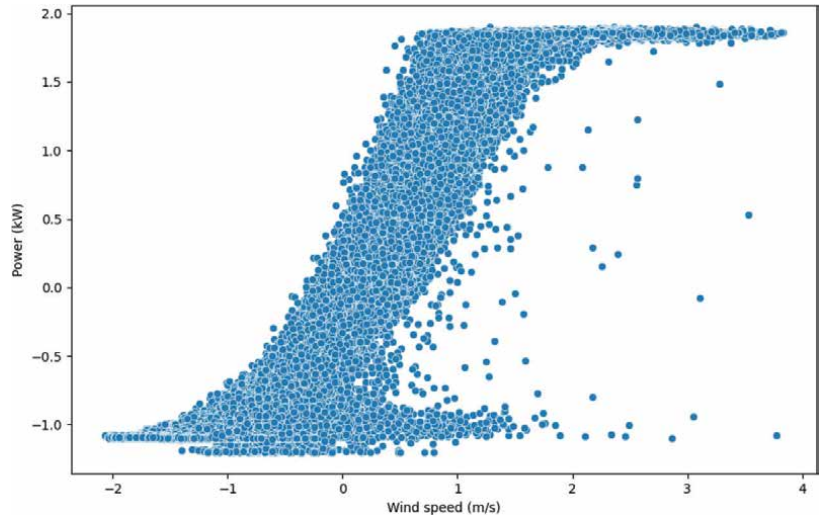

Figure 1.
Power curve, relating the wind speed and the power generated by the turbine.

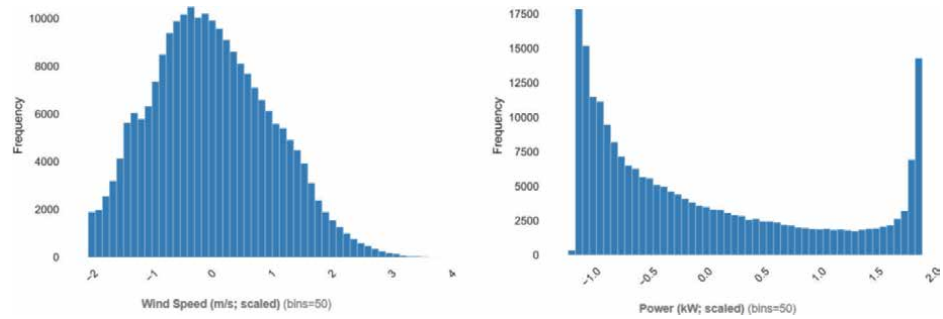

Figure 2.
Wind speed (left) and power (right) histograms.

output. An anomaly will be defined by real data escaping the prediction intervals computed by the regression and posterior techniques with statistical significance.

Once the anomalies in the regression of the power have been computed, this anomalous data will be included in the 1D-CNN as features too, simply as the deviations between registered and predicted power for each sample. As we aim to predict a failure with an anticipation that is suitable for performing early maintenance, this is a very complicated task, which we will try to compensate with feature engineering by adding rolling averages of the different original features to the data, in an agnostic manner. The time windows will be considered hyperparameters and these newly engineered features will be created anew for each hyperparameter optimization step, whereas the original features will be a constant during training of the CNN.

2.2 Deep learning models

Our first goal is to find deviations from a data-driven regression of the power prediction and the real power output feature registered by the SCADA monitoring system. The next step will be to study these deviations statistically and determine

whether their appearance is significant as the time before a fault gets shorter. Finally, we will build a classification model of the data samples for pinpointing an impending downtime of the wind turbine with sufficient anticipation. For succeeding in all these tasks, we will make use of three different neural network architectures. The power regression will be performed through two different approaches, a deep neural network (NN) that is made by a succession of Linear and Dropout layers (with ReLu activation layers within) and by using a BiLSTM with dropout, to check if the inclusion of memory cell can improve the regression even if we are not interested in using previous power predictions (autoregression). Power will always be our dependent variable. In a sense, we are extending the "Power curve" concept of computing power from wind speed but with the corrections of the other features included. As we mentioned before, creating the training dataset for this regression implied cutting out many samples, which could go against the philosophy of using recurring neural networks. However, a good performance in the first simple NN attempt would make these cuts irrelevant.

The architecture of the simpler regression model is built using PyTorch [29] ModuleList class, which allows us to build the NN with generality, determining the hyperparameters set to define it by using Bayesian optimization with Weights & Biases [30]. The BiLSTM will be defined using the LSTM class from PyTorch, with dropout and bidirectional parameters set as true. These NNs will have the following hyperparameters:

Both architectures:

- Dropout probability.
- Hidden layer size.

ANN:

- Number of fully connected layers.

BiLSTM:

- Number of recurrent layers.
- Tensor time window dimension size.

Other hyperparameters related to their training are:

- Batch size.
- Learning rate.
- Separation between well-behaved data and faults (in days).

As previously shown, Power is not Gaussian distributed. This may affect our predictions as their deviations from the real data may not be normally distributed either. However, we need them to be if we want to spot anomalies in the prediction. Thus, we will establish a custom metric that ensures this requisite is met, with the following definition:

$$metric = \left(r^2\right)^{100 \times Validation\ loss} \tag{1}$$

where r^2 is the coefficient of determination in a Q-Q plot representing the deviations on the prediction with respect to the real "Power" feature in the well-behaved validation dataset against the 45° line. The data included in this Q-Q plot is extracted by modifying the StatsModels library [31] so as to retrieve the slope and ordinate at the origin of the "s" line when building the plot. By requesting the maximization of this custom metric, since $0 < r^2 < 1$, we are ensuring that the prediction errors in the normal dataset follow a Gaussian distribution and, at the same time, NNs that present a high validation loss are penalized, with the loss being the Mean Square Error.

After the optimization of these NNs, a prediction interval based on the standard deviation of these prediction errors will be computed for both the full validation and test sets. Then, a t-Student distribution prediction interval based on 10 runs with the Dropout layers in training mode (at evaluation time) and a full Monte Carlo prediction interval with 100 runs of Dropout in training mode will be added. These last two intervals have the advantage of changing sample by sample, instead of being completely general as in the case of the standard deviation. We will define all three prediction intervals at a 95% confidence level, as we want to prove there is statistical significance in when these anomalies appear with respect to a fault.

ANOVA tests will be performed on both the full validation and test datasets, with the hypothesis of anomalies having different distributions according to the Remaining Useful Life (RUL) or time-to-fault with respect to the total anomalies recorded for a time series slice between two faults and the total samples contained in said slice.

Regarding the classification task, our choice for performing logistic regression on the data as normal or prefault has been a Tensorflow [32] based CNN architecture. Since we are aware of the challenge inherent to fault forecasting, we decided to directly implement an architecture that gets higher features but, otherwise, it is quite simple: we will have four one-dimensional convolutional layers with a max pooling layer after each two of them. We will train minimizing validation loss (binary cross-entropy), despite it not being the focus of interest in our classification though.

We will evaluate a list of metrics. The most usual in classification tasks: precision, recall and f1-score, will of course be evaluated, according to:

$$\begin{gathered} Precision = \frac{TP}{TP + FP} \quad Recall = \frac{TP}{TP + FN} \\ F1 - score = 2\frac{Precision \times Recall}{Precision + Recall} \end{gathered} \tag{2}$$

where TP, FP and FN are "True Positives", "False Positives" and "False negatives" correspondingly.

However, our hyperparameter optimization goal will be to maximize the Matthews correlation coefficient (MCC), since it is more complete by taking into consideration "True Negative" (TN) predictions, as shown in Eq. 3. This is a very appropriate metric four our problem, as we have no previous knowledge of the correct labeling, and we are facing a dataset that can become greatly imbalanced towards the well-behaved or normal label of the turbine.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3}$$

3. Results

3.1 Regression with a deep NN

After 100 runs of a Sweep (term used in Weights & Biases for a search in the hyperparameter space) with Bayesian optimization, the value obtained for the metric shown in Eq. 1 is *metric* = 0.98385. This metric has the double goal of favoring models that are statistically appropriate for anomaly extraction and fit correctly the data. Thus, a metric this close to 1 proves that a NN without a memory cell can fit the multivariate Power curve excellently. In **Figure 3**, we present the Q-Q plot showing how the deviations of our regression from the real "power" values belong to a Gaussian distribution (computed with the well-behaved data of the validation dataset).

Since the model has shown a good fit between Gaussianity and prediction deviations, the next step is to study the whole validation dataset, which includes data close in time to faults, as we want to prove that these data present anomalies. Our statistical study will study the deviations from data and prediction in the following ways: using a Monte Carlo dropout with 100 iterations and establishing a threshold of 1.96 times the standard deviation (95%) of the predictions for each sample; then computing a smaller Monte Carlo dropout sample of only 10 iterations and obtaining the t-Student distribution at a 95% confidence level and using 1.96 times the standard deviation of the global error-in-prediction distribution. These three strategies are presented as they represent and increasing speed of computation. A Monte Carlo dropout consists of setting the dropout layers of our NN architecture in train mode and perform a number of predictions for every sample, which can be quite slow. Therefore, it is interesting to find a lighter process to find an anomaly, such as computing the confidence interval defined by the mentioned t-Student distribution (Eq. 4) with a very limited Monte Carlo sampling.

$$CI^{\bar{x}}_{t(0.95)} = \bar{x}_{prediction} \pm t_{\alpha/2,n-1} \times s\sqrt{1+\frac{1}{n}} \tag{4}$$

where s is the standard deviation of the n samples used (10 in our case) and t is the pth percentile of the t-Student distribution with n-1 degrees of freedom.

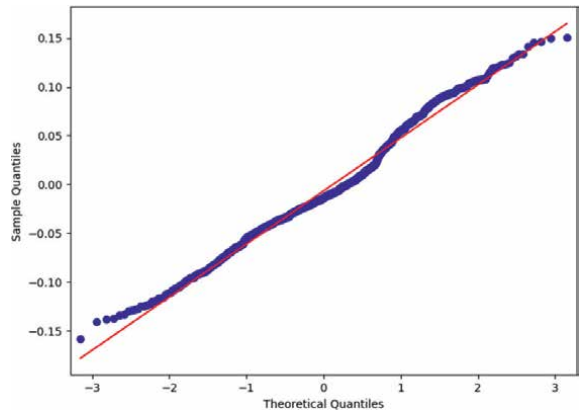

Figure 3.
Q-Q plot of the deviations between prediction and real data in the well-behaved wind turbine part of the validation dataset.

Anomaly Method	One-way ANOVA result	p-value
Monte Carlo (100 iter.)	Positive	0.143
Monte Carlo (10 iter. t-Stud.)	Positive	0.143
Standard deviation	Positive	0.14

Table 1.
ANOVA results the hypothesis of anomalies appearing when time is closer to the fault.

Any prediction exceeding the high or low limits established by these methods will be considered an anomaly according to those particular statistics. Once these anomalies have been pinpointed, our interest moves towards determining if these anomalies arise significantly as the time-to-fault reduces. We present the results of these ANOVA tests in **Table 1** with their statistical significance.

As we can see, the deep NN is successful enough both for performing a regression of the power output accurately and to find statistically significant anomalies. Let us compare now these results with those obtained by training the BiLSTM for regression. This comparison is relevant as both NNs differ greatly in the training and prediction times, being the BiLSTM much slower and with greater need of resources. First of all, we present in **Figure 4** the Q-Q plot of the best model obtained after the Bayesian optimization of the hyperparameters. This model has a metric (defined by Eq. 1) value of *metric* = 0.9732, which is still a good result for our purposes though worse than the previously obtained, with the drawback of going through a much slower training. As seen in the figure, the Q-Q plot does not fit that well the line of gaussianity in the deviation between prediction and real value.

We reproduce the one-way ANOVE tests the same way as with the previous architecture, presenting the results in **Table 2**.

This time, results are not favorable to the null hypothesis. The metric and the Q-Q plot were not as good as with the previous architecture, which may make the anomalies less reliable than those predicted by the simpler ANN, since the deviations in the validation split that only contains good behavior of the turbine are not a good fit into a Gaussian distribution, which is required to obtain reliable anomalies. It is

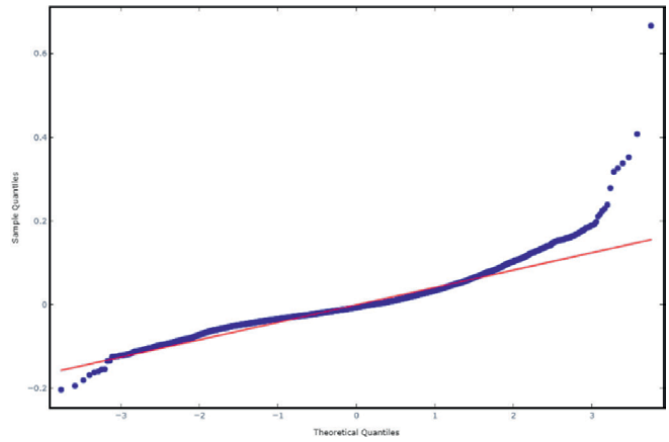

Figure 4.
Q-Q plot of the deviations between prediction and real data in the well-behaved wind turbine part of the validation dataset for the BiLSTM.

Anomaly Method	One-way ANOVA result	p-value
Monte Carlo (100 iter.)	Positive	0.018
Monte Carlo (10 iter. t-Stud.)	Positive	0.018
Standard deviation	Positive	0.018

Table 2.
ANOVA results the hypothesis of anomalies appearing when time is closer to the fault for the BiLSTM architecture.

also relevant to remind the computational semantics of data preprocessing at this stage, as the well-behaved data isolation for the training split causes cuts that can affect the memory cells of the architecture. Along this research, LSTMs have proven difficult to manage in terms of reproducibility as determinism is difficult to achieve and we included dropout to have the chance of doing the Monte Carlo sampling of predictions.

Nevertheless, these results are motivating if contradictory, which suggests the need for a way that arbitrates if it is possible to predict a fault and arranging maintenance with enough anticipation. This is where the 1-dim CNN enters. To recap the previous discussion, this is a challenging dataset where feature engineering is not an easy task and "Power" is dominated by the "Wind Speed". The use of convolution takes care of part of the feature engineering and the rolling averages with time windows defined by hyperparameters (a different one for each of the original features) ensure a non-biased feature extraction. This agnosticism is also represented in the CNN architecture, where kernel sizes and filters are defined as hyperparameters too. This search of the hyperparameter space has a dimensionality too high for Bayesian optimization to work, so we will perform an extensive random search.

Since one of our purposes is to prove the convenience of the MCC metric in classification tasks, our figures will present MCC and f1-score, showing that the latter can be in a range that is considered good despite the former being too far from 1. The Matthews correlation coefficient can range from $-1 < MCC < 1$ and close to zero is a bad fit, though it is considered a good metric when $MCC > 0.5$ (negative numbers mean that there is anticorrelation). We show the values of MCC and f1-score in **Figure 5** according to the time-to-fault, which we will plot as the number of samples labeled as 1 (we are performing logistic regression) from the fault backwards in time, which is the anticipation we seek for maintenance.

There are interesting results in this plot. There is a curve where most of the models fall and show a steady increase in both metrics. This increase is explained as more data samples are labeled as 1 or "Prefault", which biases the model making it seemingly more accurate. From the different outliers to this curve, one is very interesting, as it greatly exceeds the curve at an interesting time-to-fault. The metrics for this particular hyperparameters are:

MCC = 0.6418

F1-Score = 0.8701

Time-to-fault = 197 samples (~1 day 9 h).

It must be said that at this time-to-fault the labeling of the whole dataset is very balanced, being the "prefault" (1) samples around a 48% of the data. As it has been a very specific result without nearby hyperparameter space realizations with similar metrics, we then fix the time-to-fault and scan the remainder of hyperparameter space to determine if there are more models converging into the metrics found. The

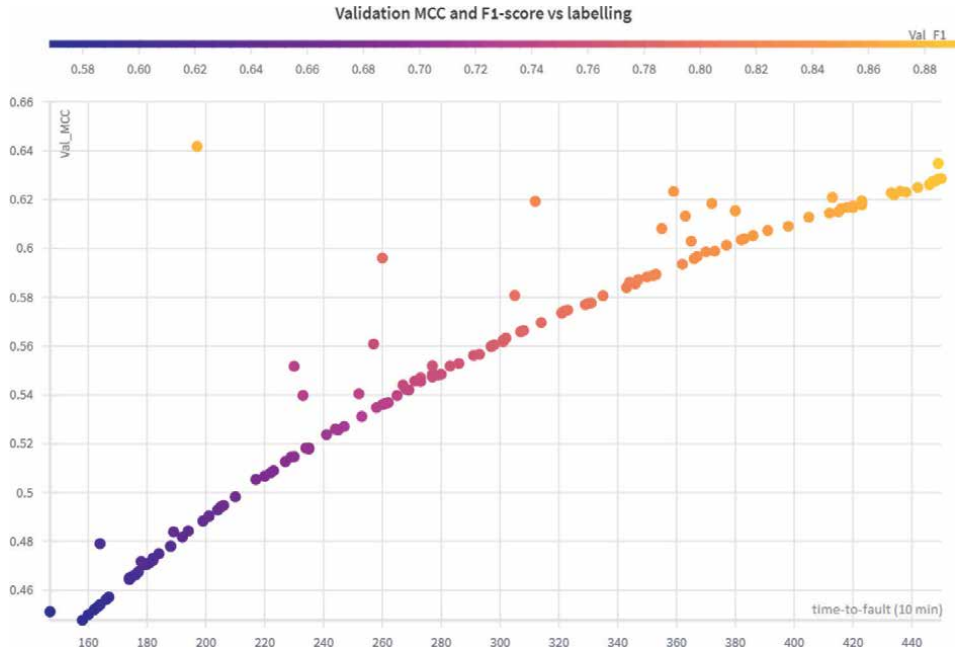

Figure 5.
MCC (left) and F1-score (color) with respect to time-to-fault (as samples labeled 1).

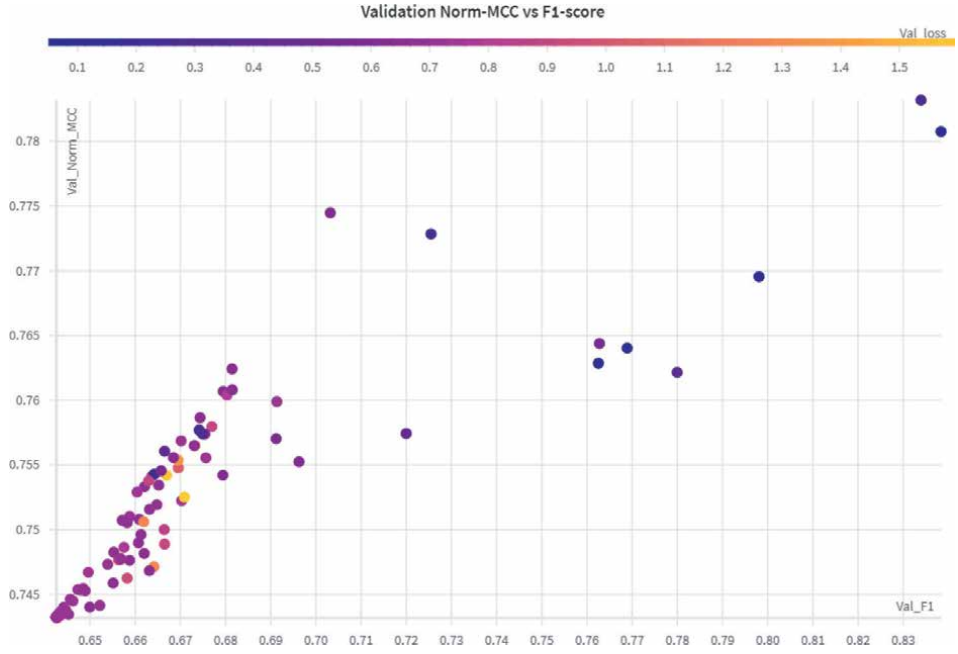

Figure 6.
MCC with respect to the F1-score for a fixed time-to-fault of 197 samples (colors represent validation loss).

results, shown in **Figure 6**, prove that after 400 random hyperparameter runs, a good f1-score is commonly achieved but it is indeed complicated to reach a good MCC metric (worse metrics than shown are cut from the figure).

Thus, it is proven that, despite the difficulty of this endeavor, it is possible to train a 1-dimensional CNN to reliable predict wind turbine faults causing downtime and schedule maintenance with at least more than a day of anticipation. In **Figure 5** we can see other time-to-fault values that are promising too, though it is understandably more complicated to predict an error the further back we move on time. For this purpose, it is highly recommendable to use the Matthews correlation coefficient instead of relying solely in the f1-score, as it is a more complete metric including True Negative samples in its computation.

4. Conclusions

Wind turbine datasets entail a high complexity for succeeding in the task of predictive maintenance. Through this chapter, we have proven that an artificial neural network can be built so as to train a regressor for the power output of the wind turbine according to the other features, where the main influence is the wind speed traditionally, as in theoretical power curves. The smart definition of metrics is the best ally to obtain a model that fits the target variable with high accuracy and can be used for computing anomalies, which requires deviations in the prediction of "Power" to fall in a Gaussian distribution for the wind turbine regime considered normal or without any fault in the time vicinity. Besides, we have shown that training a LSTM increases the difficulty of achieving these goals, which requires more computing time and resources to achieve a subpar result compared to that of the simpler NN architecture.

In addition to the regression of the power, we have developed a one-dimensional CNN architecture capable of, after an extensive hyperparameter optimization, classify any new registered data sample as a normal state or indicative of an impending fault that will cause downtime, with at least 1 day and 9 hours of anticipation. This was our main purpose, and for its achievement it has been necessary to solve both feature engineering through convolution and, as the original data is not labeled (only faults are annotated once they happen), finding the correct annotation of samples through the optimization with a powerful metric that is robust against class imbalance, such as the Matthews correlation coefficient.

To sum up, the problem of predictive maintenance without the aid of domain expertise or annotated training data can be solved with a patient hyperparameter optimization and the evaluation of strategic metrics powerful enough to train our neural networks correctly for the task.

Acknowledgements

The authors thank Enterprise Ireland and the European Union's Horizon 2020 research and innovation programme for funding under the Marie Skłodowska-Curie grant agreement No. 713654.

Conflict of interest

The authors declare no conflict of interest.

Author details

Cristian Bosch* and Ricardo Simon-Carbajo
CeADAR, University College Dublin, Dublin, Ireland

*Address all correspondence to: cristian.boschserrano@ucd.ie

IntechOpen

References

[1] WindEurope Business Intelligence; Wind energy in Europe in 2018. Trends and Statistics. windeurope.org. Feb 2019

[2] Taylor M, Ralon P, Al-Zoghoul S, Jochum M, Gielen D. Renewable Power Generation Costs in 2021. IRENA; 2022

[3] Feng Y, Tavner PJ, Long H. Early experiences with UK round 1 offshore wind farms. Proceedings of the Institution of Civil Engineers - Energy. 2010;**163**:167-181

[4] Siami-Namini S, Tavakoli N, Namin AS. The performance of LSTM and BiLSTM in forecasting time series. IEEE International Conference on Big Data (Big Data). 2019;**2019**:3285-3292

[5] Bosch C, Simon-Carbajo R. Machine learning for wind turbine fault prediction through the combination of datasets from same type turbines, floating offshore energy devices: GREENER. Materials Research Proceedings. 2022;**20**:45-57

[6] Hameed Z, Hong YS, Cho YM, Ahn SH, Song CK. Condition monitoring and fault detection of wind turbines and related algorithms: A review. Renewable and Sustainable Energy Reviews. 2009;**13**:1-39

[7] Breteler D, Kaidis C, Tinga T, Loendersloot R. Physics based methodology for wind turbine failure detection, diagnostics & prognostics. EWEA. 2015;**2015**:1-9

[8] Zhao H, Liu H, Hu W, Yan X. Anomaly detection and fault analysis of wind turbine components based on deep learning network. Renewable Energy. 2018;**127**:825-834

[9] Mazidi P, Bertling-Tjernberg L, Sanz-Bobi MA. Performance analysis and anomaly detection in wind turbines based on neural networks and principal component analysis. In: 12th Workshop on Industrial Systems and Energy Technologies (JOSITE2017). Madrid: comillas.edu; 2017

[10] Felgueira T, Rodrigues S, Perone CS, Castro R. The impact of feature causality on Normal behaviour models for SCADA-based wind turbine fault detection. ICML 2019 Workshop - Climate Change, How Can AI Help; arXiv:1906.12329; 2019

[11] Shihavuddin ASM, Chen X, Fedorov V, Christensen AN, Riis NAB, Branner K, et al. Wind turbine surface damage detection by deep learning aided drone inspection analysis. Energies. 2019;**12**:676

[12] Anantrasirichai N, Bull D. DefectNET: Multi-class fault detection on highly-imbalanced datasets. IEEE International Conference on Image Processing (ICIP). 2019;**2019**:2481-2485

[13] Mokhtari A, Belkheiri M. Fault diagnosis of a wind turbine benchmark via statistical and support vector machine. International Journal of Engineering Research in Africa. 2018;**37**:29-42

[14] Zhang DH, Qian LY, Mao BJ, Huang C, Huang B, Si YL. A data-driven Design for Fault Detection of wind turbines using random forests and XGboost. IEEE Access. 2018;**6**:21020-21031

[15] Cheng FZ, Wang J, Qu LY, Qiao W. Rotor current-based fault diagnosis for DFIG wind turbine drivetrain gearboxes using frequency analysis and a deep classifier. IEEE Industry Applications Society Annual Meeting. 2017;**54**(2017):1062-1071

[16] Chauhan S, Vig L. Anomaly detection in ECG time signals via deep long short-term memory networks. IEEE International Conference on Data Science and Advanced Analytics (DSAA). 2015;**2015**:1-7

[17] Maya S, Ueno K, Nishikawa T. dLSTM: A new approach for anomaly detection using deep learning with delayed prediction. International Journal of Data Science and Analytics. 2019;**8**:137-164

[18] Bakhtawar Shah M. Anomaly Detection in Electricity Demand Time Series Data. Sweden: KTH Royal Institute of Technology; 2019

[19] Malhotra P, Vig L, Shroff G, Agarwal P. Long short term memory networks for anomaly detection in time series. In: 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges: ESANN; 2015. pp. 89-94

[20] Zhu L, Laptev N. Deep and confident prediction for time series at Uber. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW). New Orleans, LA, USA: IEEE; 2017. pp.103-110

[21] Hill DJ, Minsker BS. Anomaly detection in streaming environmental sensor data: A data-driven modeling approach. Environmental Modelling and Software. 2010;**25**:1014-1022

[22] Zheng L, Xue W, Chen F, Guo P, Chen J, Chen B, et al. A fault prediction of equipment based on CNN-LSTM network. IEEE International Conference on Energy Internet (ICEI). 2019;**2019**:537-541

[23] Al Qasem O, Akour M. Software fault prediction using deep learning algorithms. International Journal of Open Source Software and Processes. 2019;**10**:1-19

[24] He J, Wang J, Dai L, Zhang J, Bao J. An adaptive interval forecast CNN model for fault detection method. In: 15th IEEE International Conference on Automation Science and Engineering, (CASE). IEEE; 2019. pp. 602-607

[25] Guyon I, Bennett K, Cawley G, Escalante HJ, Escalera S, Tin Kam H, et al. Design of the 2015 ChaLearn AutoML challenge. In: 2015 International Joint Conference on Neural Networks (IJCNN). Killarney, Ireland: IJCNN; 2015. pp. 1-8. DOI: 10.1109/IJCNN.2015.7280767

[26] Guyon I, Sun-Hosoya L, Boullé M, Escalante HJ, Escalera S, Liu Z, et al. Analysis of the AutoML challenge series 2015-2018. In: Frank Hutter LK, Vanschoren J, editors. Automated Machine Learning. Cham: Springer; 2019. pp. 177-219

[27] Stetsenko P. Machine Learning with Python and H_2O; docs.H_2O.ai; 2020. Available from: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/faq/general.html#i-am-writing-an-academic-research-paper-and-i-would-like-to-cite-h2o-in-my-bibliography-how-should-i-do-that

[28] Stetsenko P. Machine Learning with Python and H_2O. 2020. Available from: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/faq/general.html#i-am-writing-an-academic-research-paper-and-i-would-like-to-cite-h2o-in-my-bibliography-how-should-i-do-that)

[29] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An imperative style, high-performance deep learning library. In: Wallach H, Larochelle H, Beygelzimer A, Garnett R, editors. Advances in Neural Information Processing Systems 32. Vancouver, Canada: Curran Associates Inc; 2019. pp. 8024-8035

[30] Biewald L. "Experiment Tracking with Weights and Biases," Weights & Biases. 2022. [Online]. Available from: wandb.com

[31] Seabold S, Perktold J. Statsmodels: Econometric and Statistical Modeling with Python. In: Proceedings of the 9th Python in Science Conference; SciPy; 2010

[32] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. 2016. ArXiv, abs/1603.04467

Edited by George Dekoulis and Jainath Yadav

This book analyzes the application of computer science and artificial intelligence (AI) techniques in the semantics' analysis for linguistics, classical studies, and philosophy. Similar techniques can be implemented to incorporate the fields of education, psychology, humanities, law, maritime, data science and business intelligence. The book is suitable for the broader audience interested in the emerging scientific field of formal and Natural Language Processing (NLP). The significance of incorporating all aspects of logic design right at the beginning of the creation of a new NLP system is emphasized and analyzed throughout the book. NLP and AI systems offer an unprecedented set of virtues to society. However, the principles of ethical logic design and operation of primitive to deep learning NLP products must be considered in the future, even via the preparation of legislation if needed. As law applications are already taking advantage of the techniques mentioned, the manufacturers should apply the laws and the possible knowledge development of the NLP products could even be monitored after sales. This will minimize the drawbacks of implementing such intelligent technological solutions. NLP systems are a digital representation of ourselves and may even interact with each other in the future. Learning from them is also a way to improve ourselves.

Published in London, UK

IntechOpen

ISBN 978-1-83768-467-0
9 781837 684670