People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University of kasdi merbah Ouargla

College of Modern technology of Information and communication

Department of Electronics and Communication

MASTER Thesis

Field: Telecommunication systems

Subject

Diphone Embedding, a Step Towards Neural Network Speech Synthesis

Executed by:

- Hanane ABADA
- Asma ZERROUD

Publicly defended on 21 /06/2023 in front of the jury:

Dr. Nadjla BETTAYEB	MCB	Supervisor	UKM Ouargla
Dr. Djalila BELKBIR	MCA	President	UKM Ouargla
Dr. Sabra BENKRINAH	MCB	Examiner	UKM Ouargla

University years: 2022-2023

ملخص

الهدف من هذه الأطروحة هو بناء نظام توليد الكلام باستخدام الشبكات العصبية. يتكون العمل من بناء تمثيل ثنائيات الأصوات بناءً على نموذج التشفير التلقائي. لتحقيق ذلك، قمنا أولاً بإعداد قاعدة بيانات تحتوي على ثنائيات الأصوات ومعلوماتها اللغوية. ثم قمنا بتدريب جزء المشفر من النموذج. استخدمنا ثنائيات الأصوات وخصائصها الصوتية كمدخلات لتوليد تمثيل مشفر مرتبط بالمعلومات اللغوية. قدم تقييم النموذج المدرب واختباراته نتائج مقبولة مع معدل توقع صحيح يصل إلى 80%.

كلمات مفتاحية: توليد الكلام باستخدام الشبكات العصبية؛ تمثيل ثنائيات الأصوات ؛ التشفير التلقائي؛ الخصائص اللغوية.

Résumé :

L'objectif de cette thèse est de construire un système de synthèse de la parole basé sur les réseaux neuronaux. Le travail consiste à construire un encodage de Diphone basé sur un modèle auto-encodeur. Pour cela, nous avons d'abord préparé une base de données de sons de Diphone et de leurs caractéristiques linguistiques. Ensuite, nous avons entraîné la partie encodeur du modèle. Nous avons utilisé les sons de Diphone et leurs caractéristiques acoustiques en tant qu'entrées pour générer une forme encodée liée aux caractéristiques linguistiques. L'évaluation du modèle entraîné et de ses tests a donné des résultats acceptables avec un taux de prédiction correct atteignant 80%.

Mots-clés : Synthèse de la parole par réseau neuronal ; encodage de Diphone ; auto-encodeur ; caractéristiques linguistiques.

Abstract

Aiming for Neural Network Speech Synthesis system. The work of this thesis consists of building a Diphone embedding based on the auto-encoder model. To achieve that, we first prepared a database of Diphone sounds and their linguistic features. Then we trained the encoder part of the model. In which we took the Diphone sounds and their acoustic characteristics as inputs to generate an embedded form linked to the linguistic features. The evaluation of the trained model and it tests gave acceptable results with a correct prediction rate that resch 80%.

Key words: Neural Network Speech Synthesis; Diphone embedding; auto-encoder; linguistic features.

Acknowledgement

Thank God for allowing us to complete this work

We would like to express our special thanks to our supervisor **Ms Nadjla BETTAYEB**, who was has been a kind blessing with her guidance, continuous support and valuable advices, and for offering the spirit of motivation and optimism that helped us achieve our highest potential.

Also, we thank Mrs **Djalila BELKBIR** and Mrs **Sabra BENKRINAH** for accepting being jury members and judge our work.

In addition, we would like to express our gratitude and thanks to all the member of the electronics and communication department, Kasdi Merbah Ouargla University. Our university experience was full of challenges and successes, but thanks to their unlimited support and valuable guidance, we were able to make remarkable progress and reach new levels of success.

Finally, to those who helped us and contributed in encouraging us, whether from near or far. Here you find, an expression of our sincere thanks.

Dedication

I thank God Almighty first and foremost for the great grace that He has bestowed upon me, then I thank those who favored them. My beloved parents do not cease to me for all their efforts from the moment of my birth to these blessed moments.

I dedicate this humble work and my graduation to My dear father **Abd el kader**.

To the love of my heart, **my mother**, who has been like the shade of a tree protecting me from the hardships of life.

To the soul of my beloved **grandmother**, who was waiting my graduation day but she couldn't witness it.

My siblings: **Ahmed Yassine**, **Samir**, **Malaak Ar-Rahman**, and **Anfal**.

To those who stood by my side throughout my academic journey, like a supporting mother, my Aunt **Dr. Ouasila Abada.**

My colleagues **Ramzi Saidi** and **Abd el Majid Boudjemaa**, and my dear uncle **Omar Kafi.**

To the sister who supports me in difficult times, my beloved cousin, Professor **Soumia Abada**.

My cousins **Anouar Abada** and **Tareq Bahou.**

My teacher from whom I learned love of work, **Mr. Djamal Medjoudj**.

My best friend and source of happiness, **Radja Djoumana**.

My friend and companion, **Khadija**.

My colleagues who worked with me on this project, the kind-hearted **Asma Zeroud**, and my dears, **Isra** and **Fatima Zahra**.

To my entire family and friends.

Thank you.

Hanane Abada.

Dedication

Five years of hard work and many obstacles, can't be expressed in words. Even if it is too many, it will not narrate the joy, sadness and love we experienced.

So, thanks to God for his kindness and mercy.

I dedicate this Humble work to

my father ***MASOUD***

who always pushed me to achieve my dreams and believe in my abilities with his encouragement and support, for being the rock of our family

And to you, O Lady of the universe, in my eyes to my angel in life, you are the light of our home my mother: ***BOUSEBSI RAZIKA.***

My brothers: ***MUHAMMAD OUAIL*** and ***MUHAMMAD OUASSIM***

for their support and love.

My sisters: ***BOUTHAYNA*** and ***RAOUAN***

for all the joy and all the good memories.

My dear friends: ***SOUHILA***, ***KAOUTHAR***, ***KHAOULA***, ***BOUTHAYNA***, ***HANANE***, ***RIHAB,*** **and** ***RAMZI***

for their support and being a source of our inspiration.

To all the colleagues I met during my school years. To all my professors and teachers.

ASMA

Contents

List of Tables

.

List of Figures

List of abbreviations

ANN : Artificial Neural Networks

CNN : Convolutional Neural Networks

DEC : Digital Equipment Corporation.

DL : Deep Learning.

DNN : Deep Neural Network.

FFT : Fast Fourier Transform

FNN : FeedForword Network

FO : Fundamental Frequency.

GA : Graph auto- encoders

GCN : convolutional Graph Neural Networks

GNN : Graph neural networks

GRN : Recurrent Graph Neural Networks

GSTN : Spatial-Temporal Graph Neural Networks

HMM : Hidden Markov Model.

INNS : The International Neural Network Society

IPA : International Phonetic Association.

LPCs : Linear Prediction Coefficients.

LSTM : Long Short-Term Memory

MFCCs : Mel Frequency Cepstral Coefficients.

MUF : Mean Usual Fundamental.

NNSS : Neural Networks Speech Synthesis

RNN : Reccurent Network

SA : Standard Arabic.

SPSS : Statistical Parametric Speech Synthesis.

TTS : Text-to-Speech.

General Introduction

Text-To-Speech (TTS) is an important technology in the era of modern technology, and it is used in many different computer applications, from smart home audio applications to applications of smartphones, computers and modern robots. This technology is based on converting written text into audible speech using natural language processing techniques and voice and text recognition algorithms. Text-to-speech offers many advantages for people who have difficulty reading or who need to hear information rather than read it.

Recently, Artificial Neural Networks (ANNs) are widely used in many domains and technologies, because of their Ability to learn, to generalize, also to handle non-linear and complex relationships, Adaptability and flexibility and wide range of applications , in th field of Text-To-Speech (TTS), Neural Networks Speech Synthesis (NNSS) is now the leading technique in TTS systems.

We aim with this work to build a powerful NNSS system. By adopting the auto-encoder model, our work consists of building a di-phone embedding by training the encoder part only. This latter, is a supervised learning model that takes the sounds samples (or their acoustic features) as input, and generate an embedded representation as output. That embedding is constructed based on the sound's linguistic features. It will be used next, along with the encoder parameters, to train and build the decoder model. With the latter, we will be able to generate sound samples from their linguistic features (extracted from text).

This thesis is organized as follow:

- Chapter 1: represents speech synthesis, its important techniques and areas of use.
- Chapter 2: exposes Artificial Neural Networks, their Different models and most used Architectures (FNN, RNN, CNN, GNN, and LSTM).
- Chapter 3: presents the objective of our work, and the steps to building diphone embedding. In addition, it shows, this work evaluation and its obtained results.
- Finally, it ends with conclusion and some perspectives.

Chapter *I* :

Speech Synthesis and techniques

1 Introduction

Speech synthesis is a research field that intersects multiple domains, including electronics, computer science, linguistics, physiology, acoustics, and phonetics. In this initial chapter, our objective is to present the key concepts and knowledge required to construct a speech synthesis system in Arabic. We will provide a brief description of speech, the Arabic language, and some of their features. We will give an overview of speech synthesis and its main techniques and methods.

2 Speech

Speech refers to the manner in which we produce meaningful sounds and articulate words. It encompasses several components, including:

- Articulation: it involves the physical process of producing speech sounds using the mouth, lips, and tongue. Articulation it allows us to differentiate between words like "rabbit" and "wabbit" by accurately pronouncing sounds "r."
- Voice: That is related to the use of our vocal folds and breathe to create sounds. It encompasses variations in loudness, softness, pitch (high or low), and tone. Voice quality contributes to individuality and expressiveness in speech.
- Fluency: which refers to the rhythmic flow of our speech. It involves the smoothness and continuity of speech without interruptions or disfluencies. Some individuals may experience challenges with fluency, such as repeating sounds or pausing frequently, which can manifest as stuttering.

In overall speech combines articulation, voice usage, and fluency to enable effective communication and convey meaning. [18]

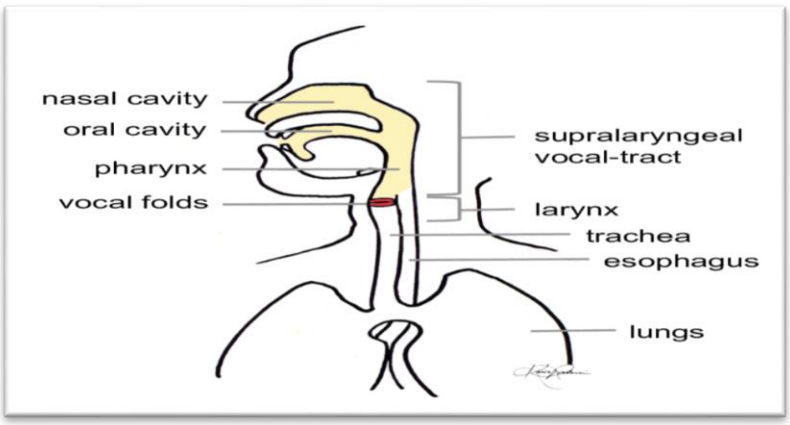


Figure1.1: A diagram of the human speech production system. [30]

Speech studying and analysis can be performed in different levels such as phonetic level and Acoustic level.

2.1 Phonetic level

Phonetics is a branch of phonology that focuses on the study of sounds and their production within the articulatory system, as well as their physical properties. [19]

2.1.1 Phonetic symbols

For most phoneticians, the symbol set of choice is the alphabet of the International Phonetic Association, known as the International Phonetic Alphabet (IPA). The latter is a set of about hundred alphabetic symbols together with handful of non-alphabet symbols (eg the length marks) and about thirty diacritics and presented, together with guidelines for their use, in the IPA Handbook. [14]

2.1.2 Definition of Phoneme

Phoneme is the smallest sound unit with no meaning, and its change may affect the word. For example in the word قَلْبٌ (a heart), the letter (ق) which is represented by the phoneme /q/ has no meaning itself, and if we replace (ق) with (ك), the meaning of the word changes to كَلْبٌ (a dog)

The word "كَتَبَ" consists of six phonemes: /k/, /a/, /t/, /a/, /b/, and /a/. [20]

2.2 Acoustic level

In acoustics, we are interested in the study of the speech signal. First, we transform it into an electrical signal, then we process it by statistical methods, there are three fundamental speech characteristics are:

- **Fundamental frequency**: is a key characteristic in speech analysis. It represents the vibration frequency of the vocal cords and varies among individuals based on the size and characteristics of their vocal cords. The fundamental frequency can be categorized into three main ranges: for men, it typically ranges from 70 to 250 Hz, for women it ranges from 150 to 400 Hz, and for children it ranges from 200 to 600 Hz. [29]
- **Duration:** The calculation of speech duration is a challenging task as it lacks a direct biological correlate, unlike fundamental frequency (FO) and energy, which depend on vocal cord tension and subglottic pressure, respectively. To determine the duration of a specific speech phenomenon, it is necessary to establish two reference points that mark its beginning

and end. In speech analysis, the segmentation process is responsible for identifying these reference points, and many modern systems rely on phonemes as the basis for segmentation. Phoneme-based segmentation plays a crucial role in accurately determining the duration of various speech phenomena. [13]

- **Intensity** or **energy**: it refers to the air pressure upstream of the larynx and it is closely associated with the sound volume of a phoneme or the magnitude of vocal cord vibrations in specific sounds. It serves as an indicator of the strength or intensity of a sound produced during speech. [29]

2.3 Speech signal analysis

Acoustic phonetics focuses on examining the physical characteristics of the sound wave as it travels from the speaker's mouth to the listener's eardrum. In this context, we will discuss some fundamental concepts related to sound waves in general, as well as the unique properties of speech signals, which are generated by the human vocal tract rather than natural environmental "noises."

By studying acoustic phonetics, we gain insights into the mechanisms and properties of speech production, transmission, and perception, allowing us to better understand the intricacies of human communication. [2]

2.3.1 Spectrogram

It is a representation of the temporal evolution of the speech spectrum, where the signal energy appears as a grey level in a two-dimensional time-frequency diagram. To get this representation, we need to take the following steps.

- Divide the signal into a series of small frames (about 10 ms).
- Apply the Fast Fourier Transform (FFT) to each of these frames.
- Position each spectrum on the Centre of each frame.
- Encode the spectrum amplitude in grey levels. [29]

The fundamental frequency exists in all voiced sounds. It corresponds to the black line superimposed on the spectrogram in VOCALAB and DIADOLAB[1] (Figure 1.5). In PRAAT[2], it appears in blue, with a specific scale on the right of the spectrogram the cumulation of the F0 values allows the extraction of the mean usual fundamental (MUF). The extraction of F0 on a siren allows the determination of the vocal range. Finally, cumulating energy as a function of F0 allows the phonogram to be drawn. [21]

[1] "VOCALAB" and "DIADOLAB" are specific software or tools related to speech analysis and acoustics.
[2] PRAAT is a specialized software for analyzing and processing sound used in linguistic research and phonetics.

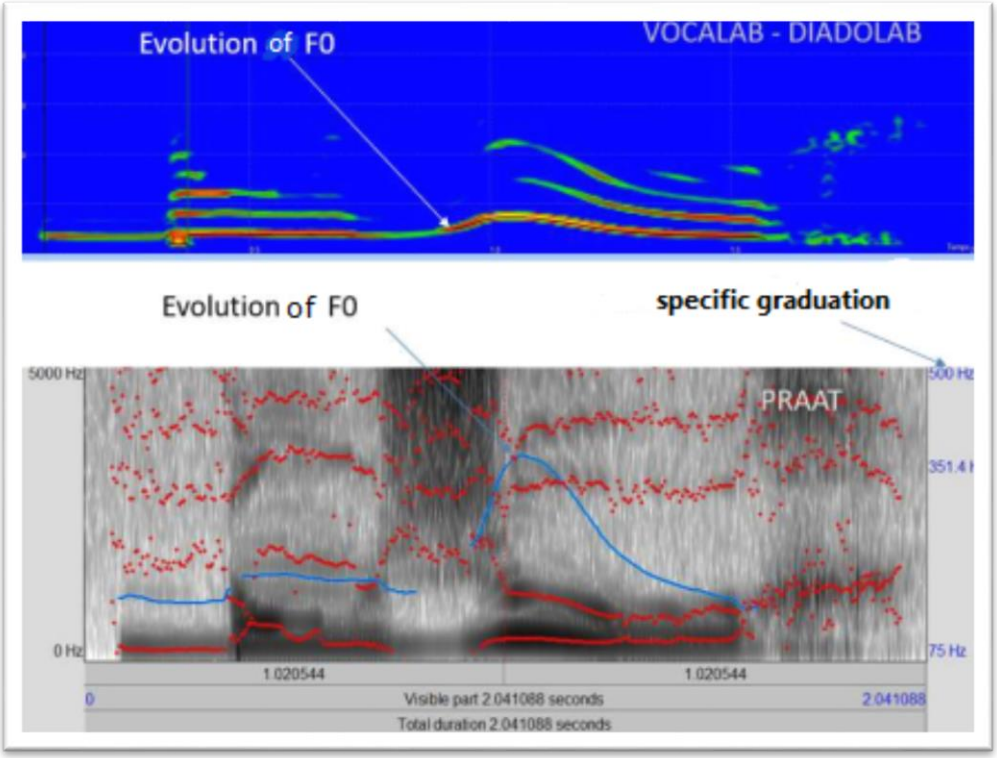


Figure 1.2: Detection of the fundamental frequency in PRAAT [21]

2.3.2 Cepstrum

The process of homomorphic transformation involves converting the speech signal from the time domain to another analogous domain. This transformation is particularly valuable for separating the effects of the vocal tract from the glottal wave, effectively isolating the fundamental frequency and the variations it undergoes within the vocal tract (referred to as harmonics).

In speech production, the resulting speech signal can be understood as the convolution of the source signal with the corresponding vocal tract filter (described by Equation 1). Thus, the need to separate these two contributions. By applying the homomorphic transformation, the convolution operation is transformed into a product, and subsequently, taking the logarithm converts this product into a sum (as demonstrated by Equations 2 and 3).

This transformation enables the analysis and manipulation of speech signals in a more meaningful and efficient manner, providing insights into the specific contributions of the vocal tract and glottal wave to the overall speech production process.

$$s(n) = u(t) * h(t) \quad \textbf{(1)}$$

With:

s(t): the time signal

u(t): the exciter signal (from the source)

h(t): the contribution of the duct.

$$S(f) = U(f) \times H(f) \quad \textbf{(2)}$$

$$\text{Log}(|S(f)|) = \log(|U(f)|) + \log(|H(f)|) \quad \textbf{(3)}$$

By an inverse transformation of the latter signal, we obtain the Cepstrum, and we will have a relationship in the time domain given by:

$$TF{-}1(\log(|S(f)|)) = TF{-}1(\log(|U(f)|)) + TF{-}1(\log(|H(f)|)) \quad \textbf{(4)}$$

Thus for one frame of a speech signal x[n] the cepstral coefficients c[n] are given by:

$$C[n] = \sum_{K=0}^{K-1} log(|\sum_{n=0}^{N-1} x[n] e^{\frac{-2\Pi i}{N} Kn}|) e^{\frac{2\Pi i}{N} Kn} \quad \textbf{(5)}$$

The independent variable of the cepstrum is nominally time, as it is the inverse Fourier transform (IFT) of a spectrum. However, it can also be interpreted as a frequency because the logarithm of the spectrum is considered as a wave. To clarify this interchangeability between the two domains, Bogert, Healy, and Tukey coined the term "cepstrum" by reversing the order of the first letters of the word "spectrum" and by analogy.

Frequency => qu f rency.

Harmonic => rahmonic.

Phase => saphe.

Filtre =>li f tre.

The analysis of this spectrum allows us to separate the contribution of the source (including the fundamental frequency) from the spectral envelope (the contribution of the vocal tract), which can be found by applying a low-pass filter. The latter can be found by applying a low-pass filter, while high-pass filtering gives us information about the pitch. If the input signal has an the input signal has a strong fundamental pitch period, it appears in the cepstrum as a peak, and by measuring the distance between the zero time and the peak time, we find the fundamental period of that pitch. [29]

2.3.3 Mel Frequency Cepstral Coefficients (MFCC)

MFCCs are cepstral coefficients represented in a so-called Mel scale. The representation in the latter is inspired by the human auditory system, which has a logarithmic sensitivity that decreases with increasing frequency. The calculation of MFCC values is done by following the steps presented in figure 1.6. [29]

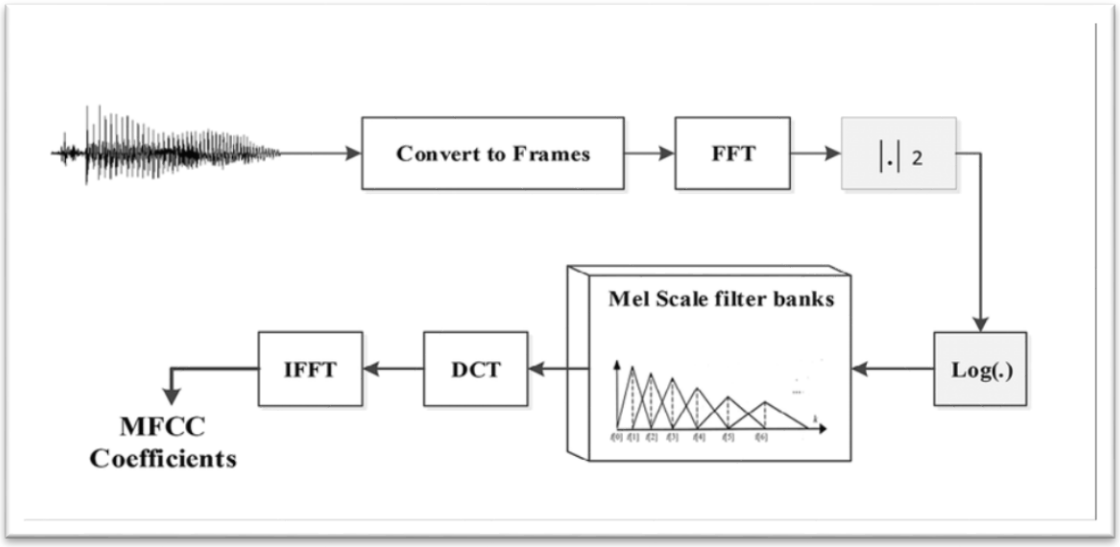


Figure 1.3: General diagram for calculating the characteristic vector of MFCC values and their derivatives. [31]

3 Arabic language

Arabic, the most widely spoken Semitic language, holds significant prominence. It serves as the primary language in 23 countries across the Middle East and North Africa. With a vast population, it is spoken by over 422 million individuals and utilized by more than 1.62 billion Muslims worldwide. In everyday communication, two main forms of Arabic are employed: dialectal Arabic, which varies among different regions, and Standard Arabic (SA), which is widely used. SA is the language of the Holy Qur'an, taught in schools, employed in literature, and used in official settings.

Like any language, Arabic operates within a systematic framework, encompassing various linguistic levels such as grammar, phonetics, morphology, and semantics. In particular, we are concerned with the phonetic and acoustic level of the language.

One distinct characteristic of Arabic writing is its direction, which is from right to left, unlike languages such as French, English, and German. [29]

3.1 The audio system of Arabic language

The Arabic language consists of 26 consonants and 2 semivowels. The Arabic alphabet comprises a set of letters, each representing a specific sound:

أ، ب، ت، ث، ج، ح، خ، د، ذ، ر، ز، س، ش، ص، ض، ط، ظ، ع، غ، ف، ق، ك، ل، م، ن، هـ، و، ي.

The audio system comprises numerous components, with the following being among the most crucial ones:

- Consonants: are all the Arabic sounds except (و, ي), so it is 26 consonant sounds.
- Vowels: Three short movements: (فتحة، ضمة، كسرة) and three long movements (المد بالألف، والمد بالواو، والمد بالياء). [27]

Figure1.4 describes the places of articulation of the Arabic sounds.

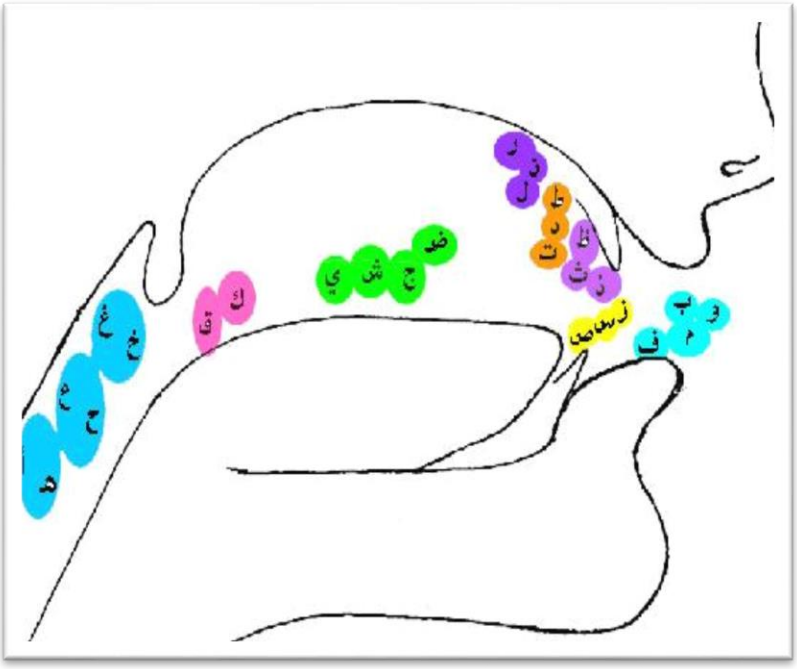


Figure1.4: the five major areas of exits of Arabic letters. [32]

3.2 Phonetic representation of Arabic letters

Phonetic transcription involves the use of phonetic symbols to represent speech sounds. The goal is to assign a specific written symbol to each sound in a spoken utterance, providing a record that allows for the accurate reconstruction of the original speech. [14] Table 1.1 presented the selected symbols from the International Phonetic Association's alphabet for the Arabic language.

Table 1.1 Arabic letters, some of their features and IPA transcription. [29]

The Arabic letters	IPA	Mode of articulation	voicing	The Arabic letters	IPA	Mode of articulation	voicing
أ	[ʔ]	Plosive	voiced	ض	[ɖˤ]	Plosive	Voiced
ب	[b]	Plosive	Voiced	ط	[tˤ]	Plosive	Unvoiced
ت	[t]	Plosive	Unvoiced	ظ	[zˤ]	Plosive	Voiced
ث	[θ]	fricative	Unvoiced	ع	**[ʕ]**	Fricative	Voiced
ج	[ʤ]	affricate	Voiced	غ	[ɣ]	Fricative	Voiced
ح	[ħ]	fricative	Unvoiced	ف	[f]	Fricative	Voiced
خ	[x]	fricative	Unvoiced	ق	[q]	Plosive	Unvoiced
د	[d]	Plosive	Voiced	ك	[k]	Plosive	Unvoiced
ذ	[ð]	Fricative	Voiced	ل	[l]	Liquid	Unvoiced
ر	[r]	Vibrant	Voiced	م	[m]	Nasal	Voiced
ز	[z]	Fricative	Voiced	ن	[n]	Nasal	Voiced
س	[s]	Fricative	Unvoiced	ه	[h]	Fricative	Unvoiced
ش	[ʃ]	Fricative	Unvoiced	و	[w]	Semi-vowels	Voiced
ص	[ʂ]	Fricative	Voiced	ي	[y]	Semi-vowels	Voiced

4 Speech synthesis

Speech synthesis or TTS (Text To Speech) is to convert any text information into standard and smooth speech in real time. It involves many disciplines such as acoustics, linguistics, digital signal processing, computer science, etc. It is a cutting-edge technology in the field of information processing, especially for the current intelligent speech interaction systems. [11]

4.1 History and TTS review

As digital signal processing technologies have advanced, the focus of speech synthesis research has shifted from intelligibility and clarity to naturalness and expressiveness. Intelligibility refers to the clarity of synthesized speech, while naturalness encompasses ease of listening and overall stylistic consistency. In the early stages of speech synthesis technology development, parametric synthesis methods were predominantly used.

In 1971, Hungarian scientist Wolfgang von Kempelen created a machine that could synthesize simple words using delicate bellows, springs, bagpipes, and resonance boxes. However, the intelligibility of the synthesized speech was very poor. To address this issue, the Klatt's serial/parallel formant synthesizer was introduced in 1980. One notable example of this technology is the DEC talk text-to-speech system developed by Digital Equipment Corporation (DEC) in Maynard, MA, USA. It provided various speech services that were understandable to users when connected to a computer or the telephone network. However, the quality of the synthesized speech remained a challenge due to difficulties in extracting formant parameters.

In 1990, the Pitch Synchronous over Lap Add (PSOLA) algorithm significantly improved the quality and naturalness of speech generated by time-domain waveform concatenation synthesis methods. However, PSOLA required accurate annotation of pitch period and starting point, as errors in these factors could greatly affect the quality of synthesized speech. Despite advancements, synthesized speech using this method still did not match the naturalness of human speech. To address this, researchers delved into speech synthesis technologies and employed statistical parametric speech synthesis (SPSS) models to enhance naturalness. Notable examples include Hidden Markov Model (HMM)-based and Deep Learning (DL)-based synthesis methods. [11]

4.2 Architecture of a speech synthesis system

A typical Text-to-Speech (TTS) system comprises two primary processing blocks: linguistic processing and acoustic processing. The linguistic processing block analyzes and structures the text to determine a coherent pronunciation mode. It then transforms the analyzed text into a sequence of symbolic descriptors that represent the target speech units. The acoustic processing block generates an acoustic signal that is adapted to this symbolic sequence.

Figure 1.7 illustrates the overall architecture of a text-to-speech system. The first two components, involving high-level processing, facilitate the transition from the orthographic representation of the input text to a phonetic representation with prosodic details. The final component encompasses the low-level processing of the synthesizer, responsible for generating the actual acoustic signal. [12]

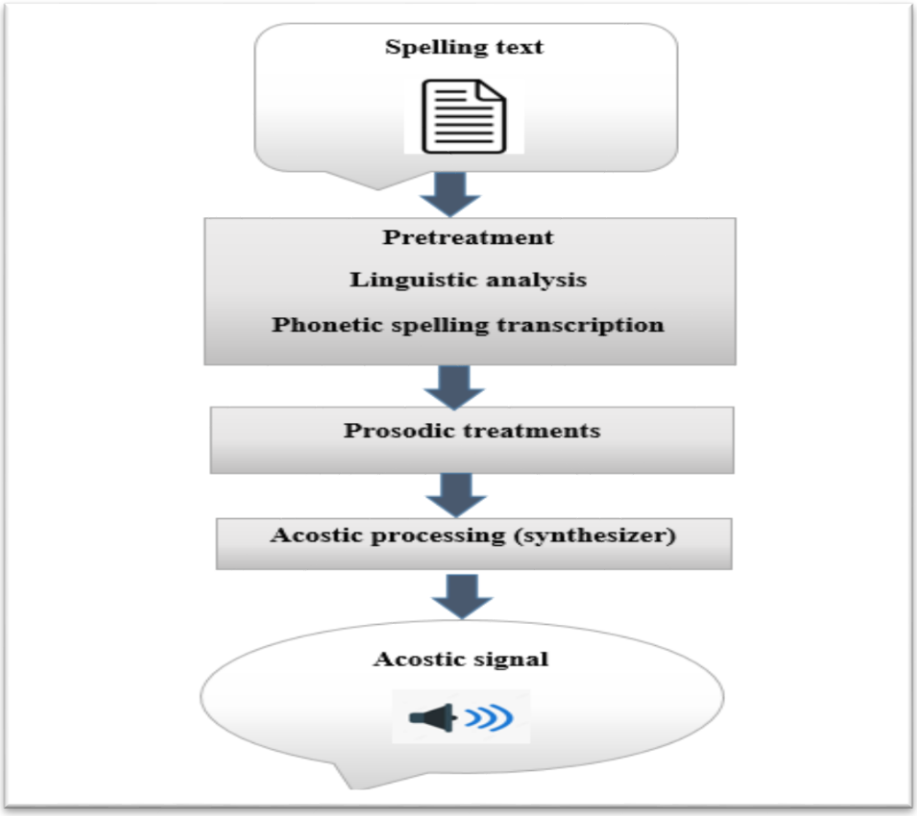


Figure 1.5: architecture of a speech synthesis system.

4.3 Speech synthesis methods

There are two main ways (parametric and Concatenative).

4.3.1 Parametric synthesis speech

The Statistical Parametric Speech Synthesis (SPSS) system consists of three essential modules: text analysis, parameter prediction, and speech synthesis.

- The text analysis module pre-processes the input text by performing tasks such as text normalization, automatic word segmentation, and grapheme-to-phoneme conversion. It converts the text into linguistic features used by the speech synthesis system, including phoneme, syllable, word, phrase, and sentence-level features.
- The parameter prediction module utilizes a statistical model to predict acoustic feature parameters, such as fundamental frequency (F0), spectral parameters, and duration, based on the output of the text analysis module. Its main objective is to estimate the acoustic characteristics of the target speech.

- The speech synthesis module employs a specific synthesis algorithm to generate the waveform of the target speech. It takes the output of the parameter prediction module as input and produces the synthesized speech accordingly.

The SPSS system consists of two main phases: the training phase and the synthesis phase.

- In the training phase, acoustic feature parameters (e.g., F0 and spectral parameters) are extracted from a corpus. Subsequently, a statistical acoustic model is trained using the linguistic features from the text analysis module and the extracted acoustic feature parameters.
- In the synthesis phase, the trained acoustic model is used to predict the acoustic feature parameters based on the linguistic features. These predicted parameters are then utilized to synthesize the speech using a vocoder, resulting in the final synthesized speech output. [11]

4.3.2 Concatenative speech synthesis

Concatenative speech synthesis involves concatenating pre-recorded and labeled speech units from a database to generate continuous speech. This method improves the naturalness of synthesized speech by selecting appropriate units based on analyzed context information. Two schemes are commonly used: one based on f (LPCs) and the other based on PSOLA. LPC-based synthesis preserves speech information and produces natural-sounding individual words. However, concatenation points, as natural speech is not simply a concatenation of isolated units affect the overall effect. PSOLA addresses this by adjusting prosody based on the target context, resulting in a waveform that maintains speech quality and conforms to prosody features. PSOLA has limitations, including sensitivity to pitch period and starting point, and difficulty in achieving smooth transitions. These drawbacks restrict its application in diverse speech synthesis. [11]

4.4 SPEECH SYNTHESIS APPLICATIONS

Current text-to-speech applications can be grouped into five broad areas:

- **Aids for people with disabilities:**

• reading screens.

• Voice communication aids for people who are dumb, laryngectomies or cerebral palsy.

• voice logs, etc.

- **Computer Aided Instruction Tools (OEAO):**

• Automatic dictation system.

• Language learning system.

- **Industrial applications:**

• Alert, site monitoring and network monitoring servers.

• Remote maintenance.

• help functions in cockpits.

• Voice verification function in editing stations (correction of proofs) or entry of written information (databases), etc.

- **Non-telephone consumer applications:**

• Home automation (alarms, domestic talking devices, etc.).

• Micro-computing (games and talking CDROMs, office automation, etc.).

- **Voice telematics:**

• Voice information servers (synthesis replacing recorded natural speech for rapidly evolving information available in text form).

• Servers for voice reading of FAX or electronic messages (e-mails).

• Automation of order taking services (mail order).

• Automation of information services (directories, company standards, etc.). [12]

5 Conclusion

This chapter introduced the field of speech, specifically focusing on speech synthesis. We began by giving a concise overview of the Arabic language, and then we interred the world of speech synthesis, examining its fundamental techniques and methodologies.

Chaptre ∏ :

Artificial Neural Network

1. Introduction

The human brain has remarkable information processing power. The nature of intelligence has always been difficult, such as the ability to observe, understand, learn, etc. Currently, artificial intelligence technologies is widely used to simulate human thinking in many application .Among all the smart technologies, Artificial Neural Networks (ANN) is a well understood and proficient data processing technology, and that we will study in this chapter.

2. Artificial neural networks (ANN)

ANN is an interconnected group of virtual neurons created by computer programs to mimic the work of a biological neuron. It can also be considered as an electronic structure that use mathematical model to process information based on the communicative method in computing.

2.1. History

Artificial neural networks have witnessed many developments throughout history. The following table presents some of main upgrades: [1]

Table 2.1: The evolution of the Artificial neuron network

Date	**Inventor**	**The most important inventions**
1890	W. James	Proposed law of process for learning on neural network (Hebb's law)
1943	J. Mc Culloch and W. Pitts	Perform the first computational operation by a neuron to simulate the biological cell model, and it is the first simple network (with logical and arithmetic operations)
1957	F. Rosenblatt	develops the Perceptron model. He builds the first neuron- Computer based on this model and applies it to the field of pattern recognition
1960	B. Widrow	develops the Adaline model (Adaptive Linear Element). In its structure, the model resembles the Perceptron
1969	M. Minsky and S. Papert	fixe the problems of Perceptron model.
1982	J. J. Hopfield	Presents a theory of Functioning and possibilities of neural networks.
1983	The Boltzmann	The Boltzmann Machine which is the first known model capable of treating the perceptron, but the practical use Proves to be difficult (the computation times are considerable).
1985	three groups of independent researchers	The gradient back propagation. It is a learning algorithm suitable for Multilayer Neural Networks (also called Multilayer Perceptrons).
1992	The Neuro-Nîmes congress	The theme of which is neuromimetic and their application

2.2. Biological and Artificial Neuron

A neuron consists of a cell body that processes information and returns the result in the form of electrical signals. The cellular body sums up the influxes that reach him, if this sum exceeds a certain threshold, it sends, itself, an influx through the axon. The axons link the neurons together, therefore play an important role in the nervous system The neuron is also made up of several branches called dendrites, which are the receptors of the neuron .

The (formal) artificial neuron is a theoretical model for processing information inspired by observations related to the function of a biological neuron. It is intended to reproduce intelligent reasoning in an artificial way. [2]

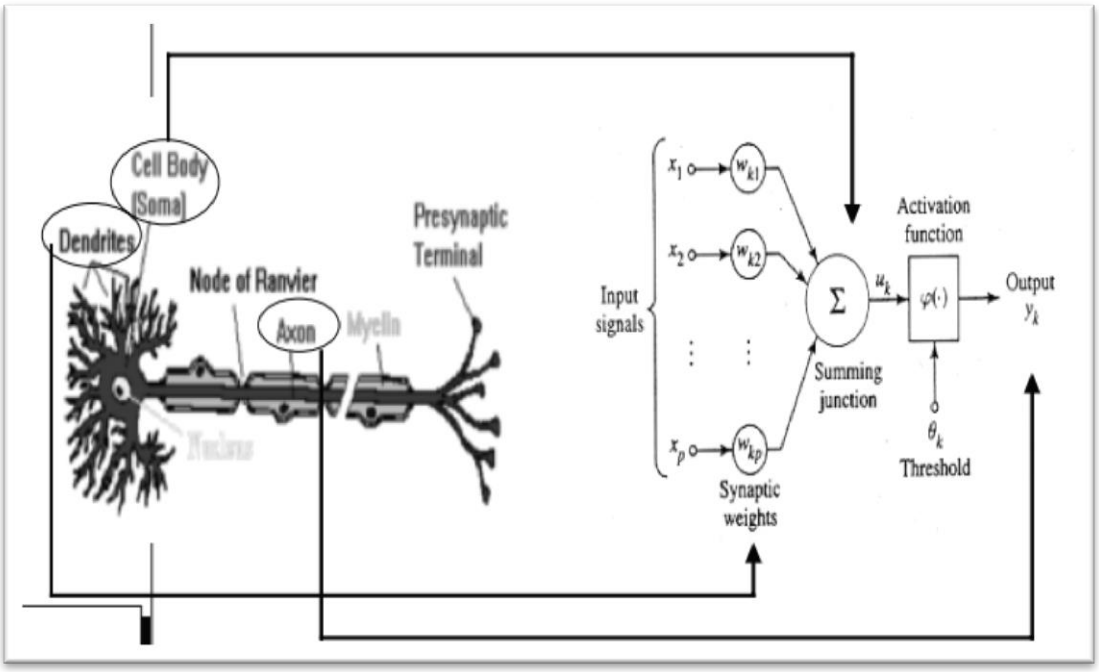


Figure2. 1 : From Biological Neuron to Artificial Neuron. [2]

2.3. Different models of neural networks

2.3.1. Monolayer Neural Networks (simple Perceptron):

It is the simplest neural network developed by Frank Ronsblatt and consists of two layers:

the inputs x1,x2,….xn , each entry has weight. The output layer is the one that gives the answer and the target of the model corresponding to the stimulus present at the input and it in terms of weight and inputs. [3]

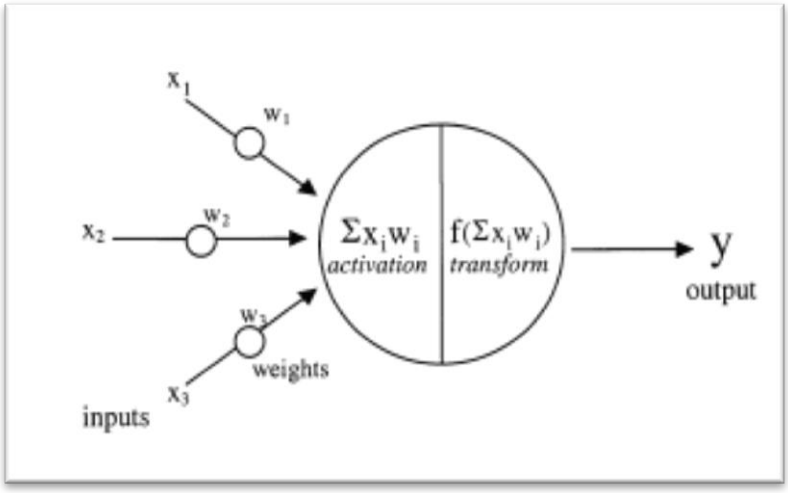


Figure2. 2: MonoLayer Neural Networks.

2.3.2. Multilayer Neural Networks (Perceptron multilayer)

It consists of multiple layers each layer is connected to the next through weight-related uses, it's basic components include: an input layer(first layer), hidden layer (these layers consist of a number of nerve units connected to each other) and output layer

Multilayer Neural Networks used to control model complexity or capacity [10]

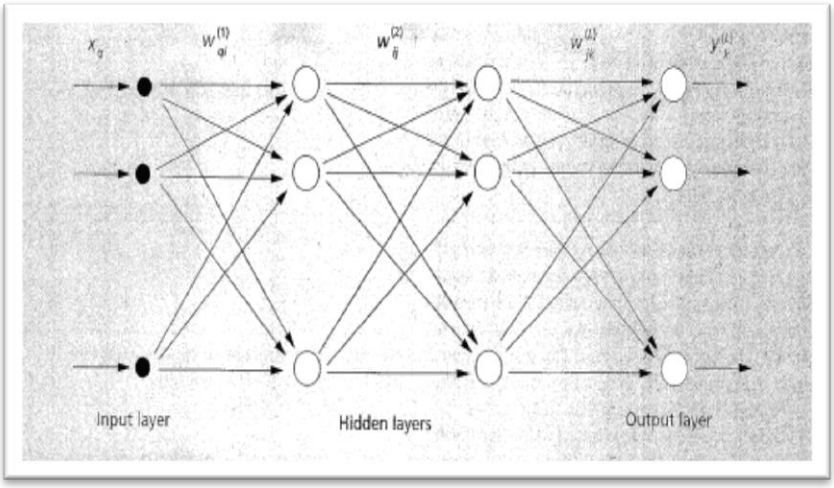


Figure2. 3: Multilayer Neural Networks. **[**10]

2.3.3. Auto-encoder

An auto-encoder is a specific type of artificial neural network used for unsupervised learning and dimensionality reduction. It has been first introduced in 1990s. It is composed of two main parts: an encoder and a decoder.

1. Encoder layer: It is the first part of the model and is responsible for converting the original data into a compressed representation. It consists of a set of neural layers that

convert the data you pass through into a lower-dimensional representation. These layers are trained to extract distinctive and important features in the data.

2. Decoder layer: This is the second part of the model and it restores the original data from the compressed representation. This layer consists of a set of neural layers that convert the compressed representation into a representation that is as similar as possible to the original data. The goal of training these layers is to achieve the best possible recovery of the original data. As present Figure 2. 4 .

auto-encoders are used for : dimensionality reduction, feature extraction, data denoising and generative modeling.[9]

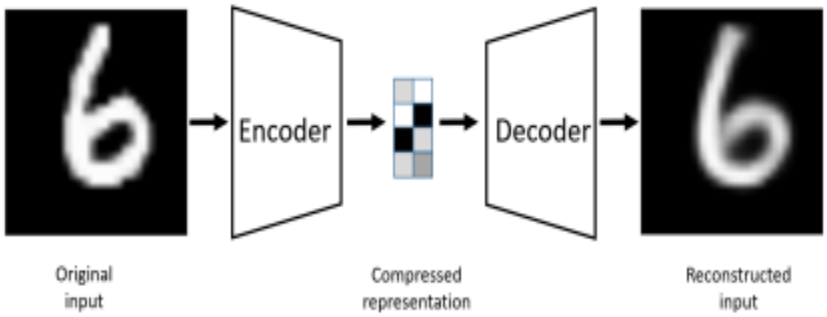


Figure2. 4: auto-encoder architecture. [9]

3. Architecture of Artificial neural networks

Artificial neural networks has 3 important elements, which are the input, output, and hidden layer, where the network begins to recognize the inputs x1,x2,….xn, then sends the signal to the hidden layer, and distributes it to the activation function according to the value of X, and each signal has a precisely defined weight, after that will be sent to the output Y.

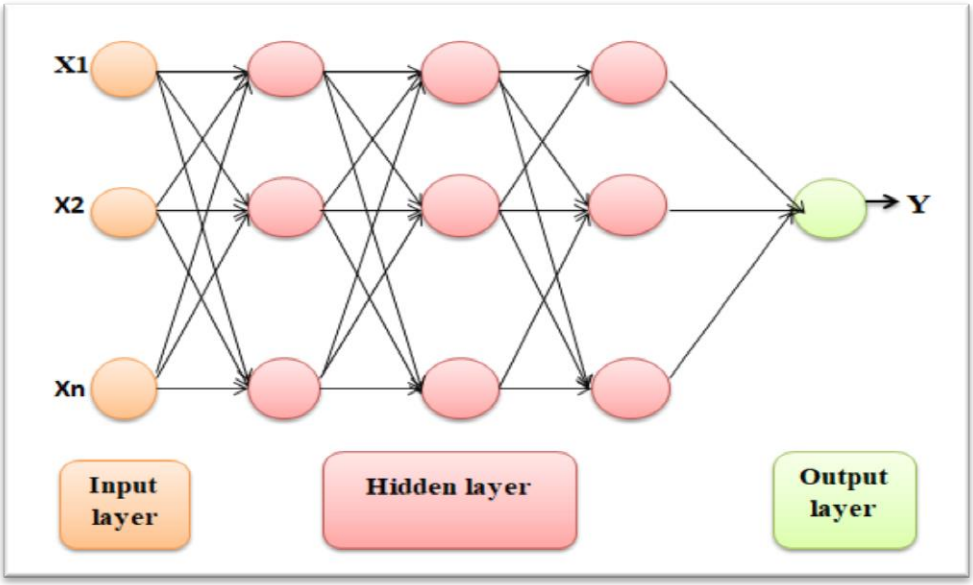


Figure2. 5 : architecture of Artificial neural networks.

Figure2.As presents the figure 2.6, different activation functions can be used depending on the network utility.

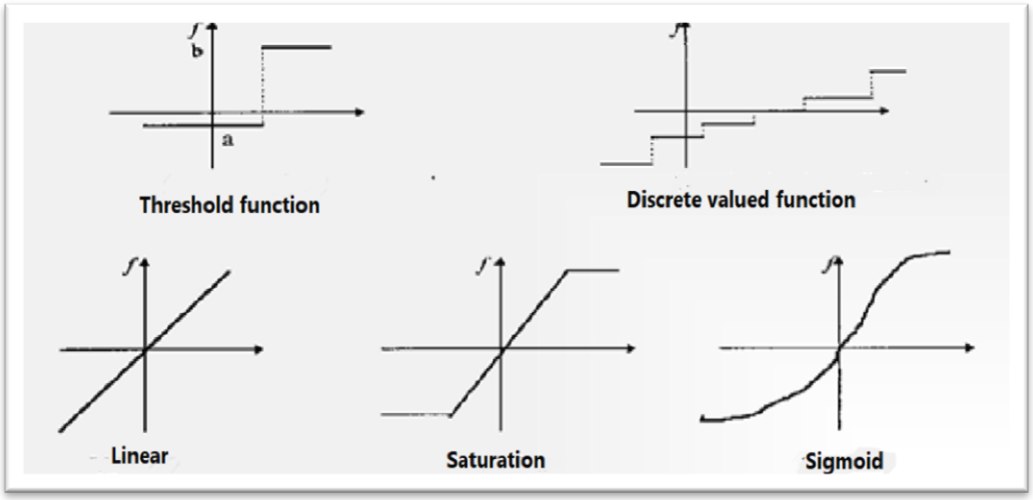


Figure2. 6: some used activation function.

Neural network architectures differ in the number, type, and arrangement of layers and computational units that make up them. These differences affect the neural network's ability to learn, analyze data, and predict results. And by choosing the appropriate architectures, functions of different complexity and power, and here some used architectures.

3.1. Feedforward Neural Networks (FNN)

FNN are composed of (figure 2.7):

a) Input layer: the number of neurons in this layer corresponds to the number of inputs to the neuronal network. This layer consists of passive nodes which do not take part in the actual signal modification, but only transmits the signal to the following layer.

b) Hidden layer: this layer has arbitrary number of layers and neurons. Its nodes i take part in the signal modification, hence, they are active.

c) Output layer: The number of neurons in this ending part corresponds to the number of output values. The nodes in this layer are active ones.

FNN has the features of :

- No feedback within the network;
- The coupling takes place from one layer to the next.
- The information flows, in general, in the forward direction. [4]

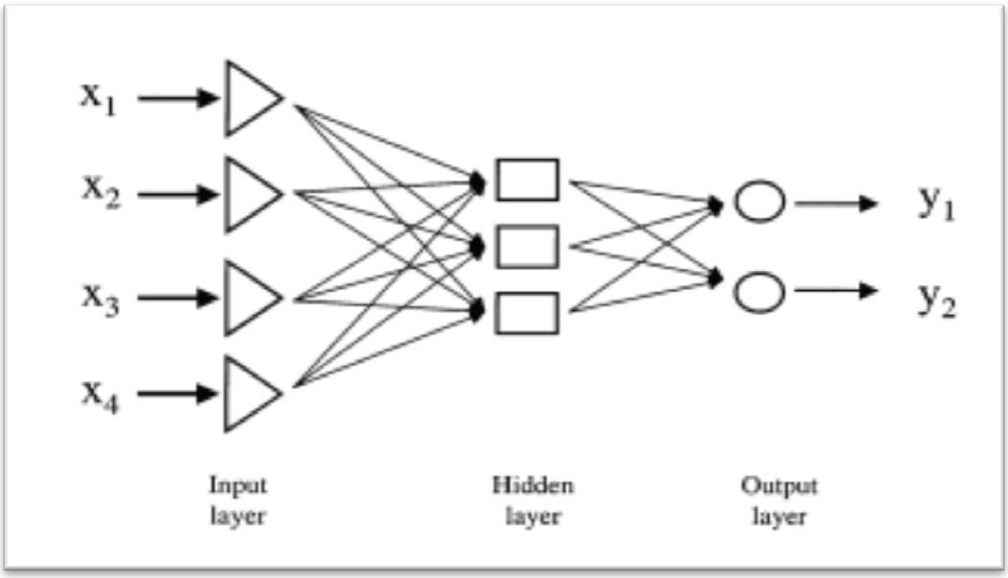


Figure2.7 : Feedforward network architecture. [4]

3.2. Feedback networks(Recurrent Network) RNN

In RNN the output of a neuron is either directly or indirectly feed back to its input via other linked neurons. Whilst there are many different types of control systems, there are just two main types of feedback control namely: Negative Feedback and Positive Feedback. Recurrent networks have a feedback where data can be fed back into the input at some point [4].

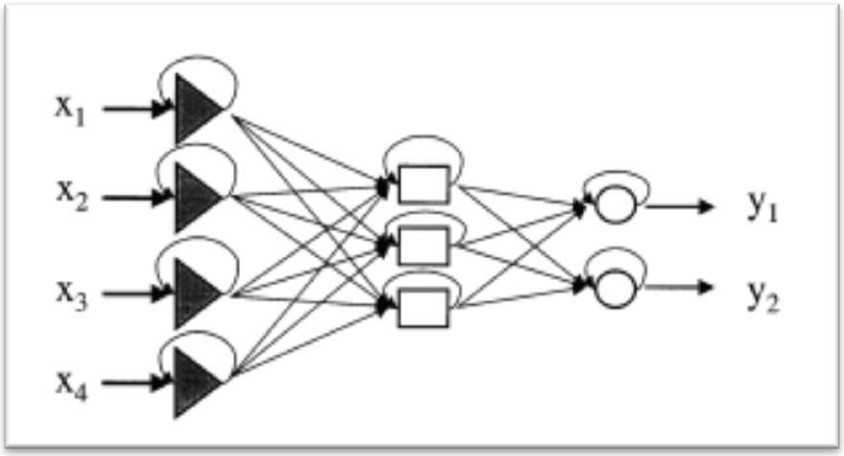


Figure2.8 : Feedback network architecture. [4]

3.3. Convolutional Neural Networks (CNNs)

CNNs are Artificial Intelligence algorithms based on multi-layer neural networks that learns relevant features from images. they are capable of performing several tasks like object classification, detection, and segmentation.

CNNs have been successful in identifying faces, objects and traffic signs apart from powering vision in robots and self-driving cars. [7]

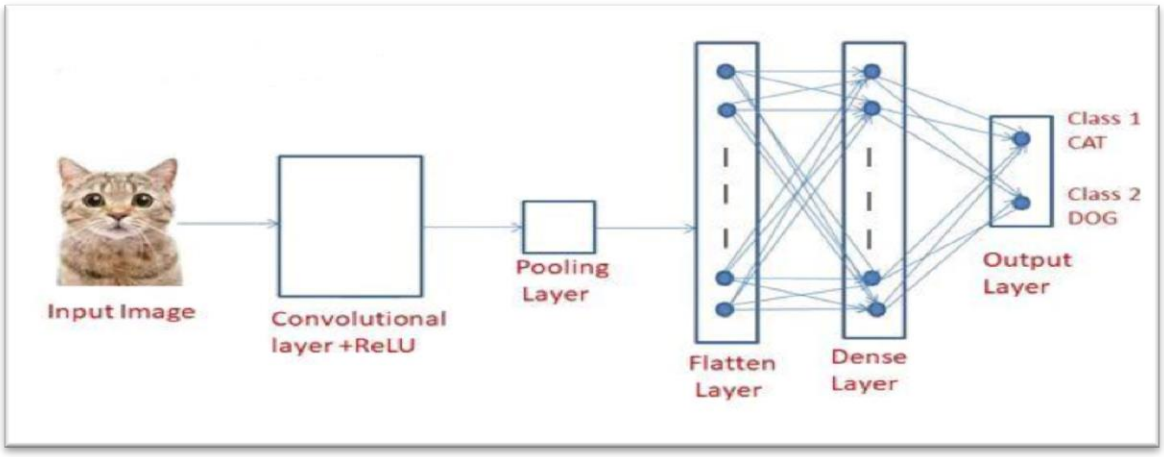


Figure2. 9: CNNs architecture. [7]

A CNN is topically composed by three types of layers:

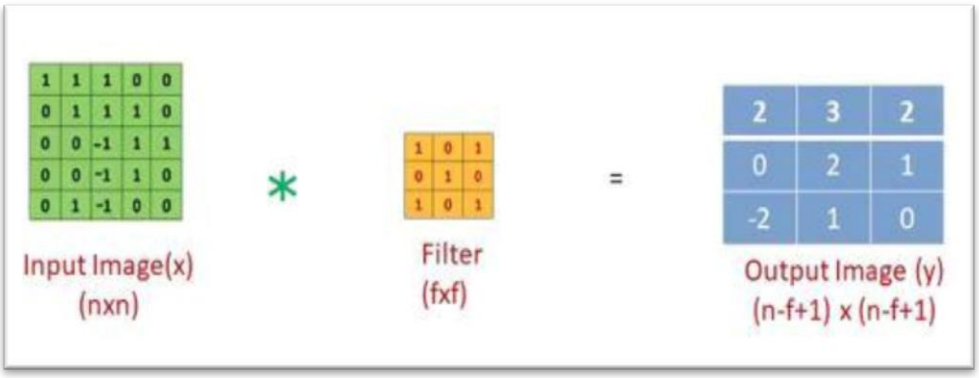


Figure2. 10 : Convolutional layer. [7]

- Convolutional Layer (with ReLU activation): The primary purpose of Convolution is to extract features from the input image. Convolution preserves the spatial relationship between input by learning input features.
- Pooling Layer Fully: Pooling layers would reduce the number of parameters when the inputs are too large. This step is also called down sampling which reduces the dimensionality of each map but retains important information. There are three types of pooling namely Max Pooling, Average Pooling and Sum Pooling.

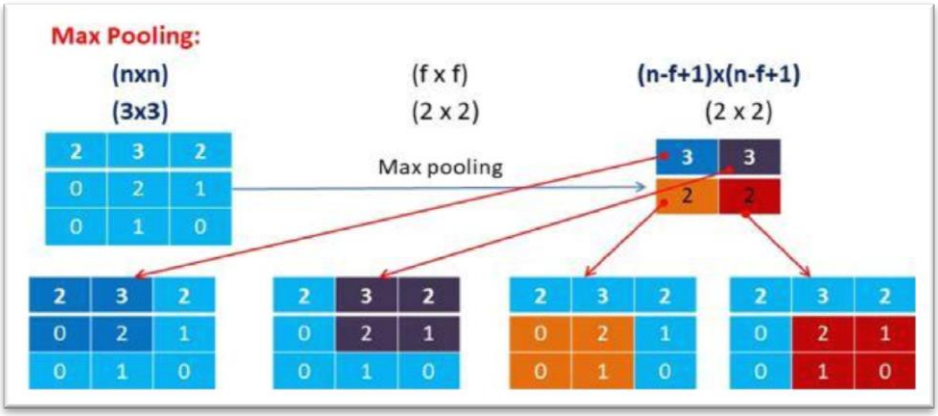


Figure2. 11 : example of max pooling Layer. [7]

- Fully Connected or Dense Layers & softmax activation

The Fully Connected layer is a MultiLayer Perceptron (MLP), composed by three types of layers: input, hidden, and output layers. The output layer has a different activation function. Usually, the softmax function is used to generate the probabilities of each category in the problem scope.

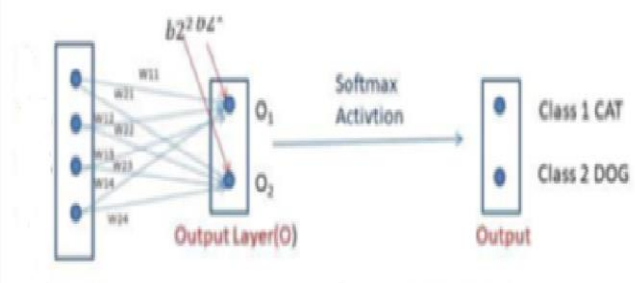


Figure2. 12: The fully connected layer.**[7]**

3.4. Graph Neural Network(GNN)

Graph neural networks (GNNs) are deep learning based methods that operate on graph domain. Based on CNNs and graph neural network business variables have been proposed to collect information collectively from the graph structure. Thus they can model the inputs and/or outputs made up of the elements and their dependencies. It's also kind of data structure which models a set of objects (nodes) and their relationships (edges)GNN is categorized into four groups: recurrent graph neural networks (GRN), convolutional graph neural networks (GCN), graph auto- encoders (GA), and spatial-temporal graph neural networks(GSTN). [6]

3.4.1. **General design pipeline of GNNs:** Generally, the pipeline contains four steps:

a) Find graph structure.

b) Specify graph type and scale.

c) Design loss function.

d) Build model using computational modules.

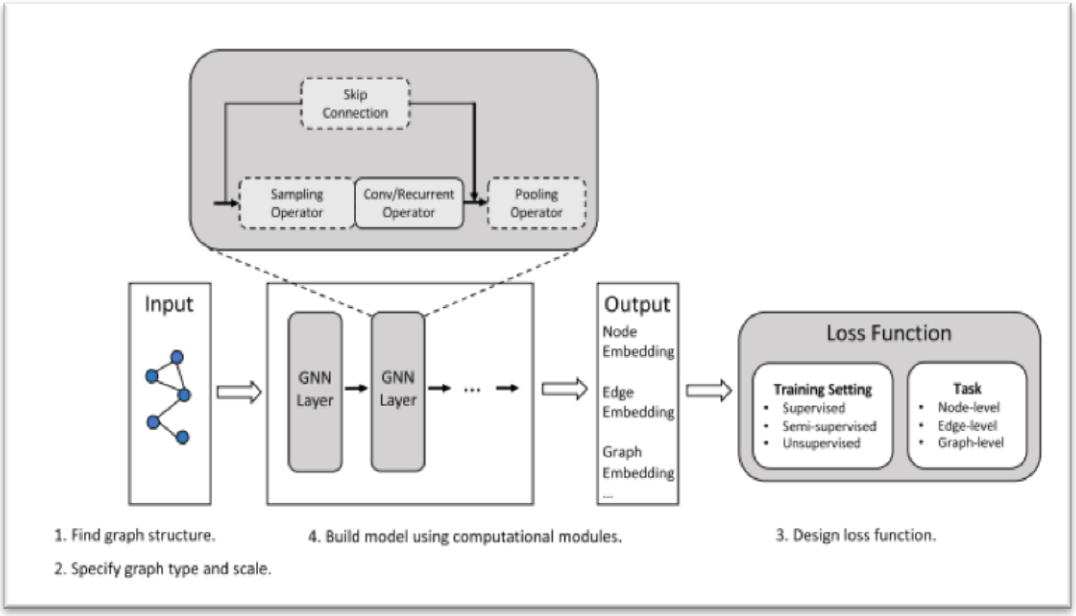


Figure 2. 13: The general design pipeline for a GNN model. [6]

3.5. Long Short-Term Memory(LSTM)

LSTM was first introduced in 1997 by **Sepp Hochrieiter** and **Jürgen Schmidhuber**. It's a type of recurrent neural network (RNN) that is specifically designed to overcome the limitations of traditional RNNs. And it's have gained significant popularity in various domains due to their ability to model long-term dependencies in sequential data and data of temporal nature. [8]

The key innovation of LSTM networks is the introduction of memory cells, which allow the network to selectivel remember or forget information over long time intervals. The memory cells are equipped with gates that flow of information, consisting:

- Forget Gate: determines which information from the previous memory cell state to discard.
- Input Gate : controls what new information is added to cell state from current input.
- Memory Update :the cell state vector aggregates the two components (old memory via the forget gate and new memory via the input gate)
- Output Gate : Conditionally decides what to output from the memory

Here are some of the common use cases and applications of LSTM:

Natural Language Processing (NLP), sentiment analysis, including machine translation, Speech Recognition, Gesture Recognition and music Generation

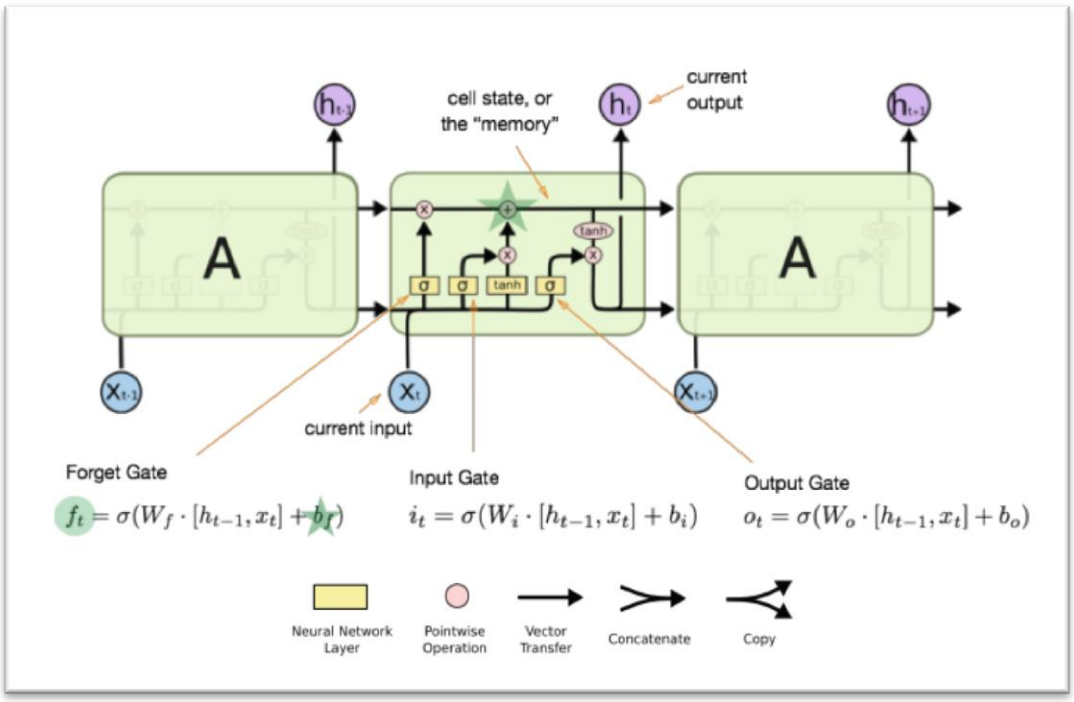


Figure 2. 14: The hidden state of a LSTM. [8]

4. Conclusion

This chapter has given us an overview of the field of artificial neural networks and some of his different architectures :Feedforward networks, Feedback networks, GNN, CNN and LSTM. This laters are counted powerful and flexible machine learning models, capable of solving a variety of complex problems. In our work we chose LSTM because it is the best to express the ability to handle sequences and data of temporal nature.

Chapter *III:*

Diphone Embedding, Evaluation and Results.

1. Introduction

This chapter is divided into two parts. the first one presents the tools and the followed steps to build a diphone embedding. While the second part shows the obtained results after evaluating the built model.

2. Model description and objective

Neural Network Speech Synthesis (NNSS) is a parametric speech synthesis that is based on training a neural network model to generate acoustic sound from some text extracted parameters (linguistic features). we chose to adopt the auto encoder (encoder, decoder) model to build such system.
In this work, we concern to train the encoding part only to generate an embedded representation of the input sound sample. We wanted to connect the resulting embedding with the sound linguistic features. So, we can use it along with the encoder model parameters to build and train the decoding part (figure 3.1). The latter will be able in the end to generate sound sample from linguistic features that can be extracted from text.

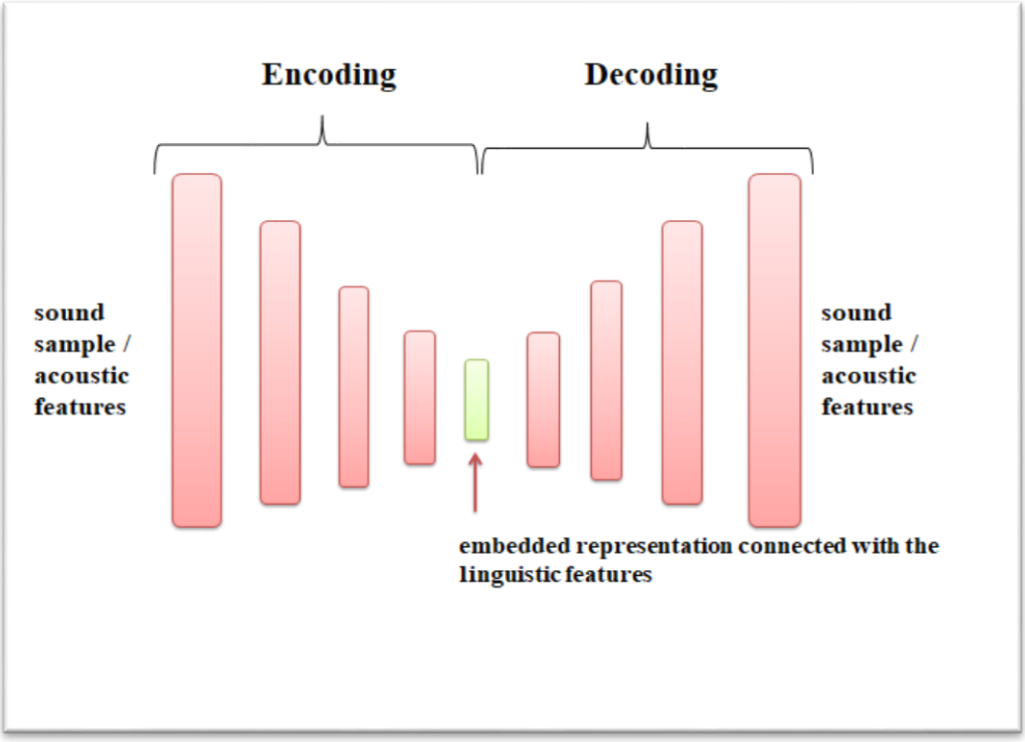


Figure 3. 1 : An auto-encoder model for speech synthesis.

3. Used tools

to build and train the proposed model we needed some Hardware and software tools consist of:

3.1. Working device

We have worked with an HP personal computer: With Processor (11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz 3.00 GHz), and RAM of 4.00 Go .

3.2. Python Programming language

Python (Figure 3.2) is a programming created by the Dutch Guido van Rossum in 1989. The name Python comes from a tribute to the television series Monty Python's Flying Circus. It is a high-level scripting language, structured in open source; the first public version of this language was released in 1991. The latest version of Python is version which was introduced on 30 August2021. Python is the most used programming language in the field of Machine Learning, Big Data and Data Science. [25]

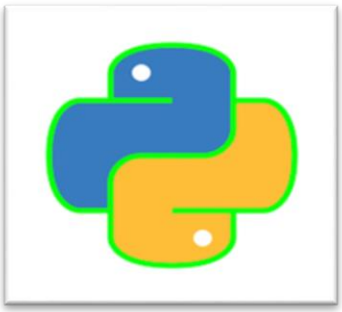


Figure 3. 2: The logo of python.

We used Python because:

- ✓ it has a simple and easy to learn syntax, that makes it ideal for beginners in programming. Also, it has a large community of developers who work together to create libraries and frameworks, as well as to solve programming problems and flexibility;
- ✓ it can be used for a variety of programming tasks, including web development, data analysis, machine learning;
- ✓ it is an open-source programming language, which means that its source code is freely available and modifiable. This allows anyone to contribute to its development and create custom

applications and large standard library**,** which provides functionality for many common programming tasks such as string processing, file manipulation, database access, etc;

- ✓ Python is relatively fast for many common programming tasks. [26][27].

Among the many libraries exist in Python, we have used the following:

- ✓ Numpy**:** is an open-source software library for programming in Python that provides tools for manipulating multidimensional arrays and performing mathematical operations on those arrays. I was created by Travis Oliphant in 2005 [28].
- ✓ TensorFlow: is an open-source software library developed by Google for machine learning and data processing. It enables the creation and training machine learning models such as DNN, CNN and RNN, as well as performing complex mathematical operations on multidimensional arrays. TensorFlow was first released in 2015 [29].
- ✓ Keras: is an open-source library for machine learning written in Python. It makes building, training, and deploying deep learning models quick and easy. Keras was created in 2015 by François Chollet, a computer engineer at Google, with the aim of providing a user-friendly interface for machine learning developers [30].
- ✓ Librosa: is a software library used for parsing and processing audio in the Python programming language. This library was developed by a student in University of **Molosa** in **Belgium**. The powerful and flexible Librosa library is designed to download and extract audio files, as well as, building blocks necessary to create music information retrieval systems [31].

3.3. Google Colaboratory (Colab):

Colab is a cloud-based integrated development environment (IDE) built on Jupyter Notebook. It allows to write and run Python code directly in the browser without the need for local setups or complex configurations. It is a powerful tool for running and developing code, suitable for tasks such as deep learning, machine learning, software development, and data analysis. Also it:

- ✓ provides features like easy importing of libraries and modules,
- ✓ allows free access to Graphics Processing Units (GPUs) to accelerate deep learning operations, and support for various environments such as TensorFlow, PyTorch, and Keras.
- ✓ provides the ability to upload and save Notebooks and relates data to Google Drive account.
- ✓ allows the step-by-step code execution and immediate results view.

4. Diphone embedding building

Our work consists of training the encoder of auto-encoder to build an embedded representation of our audio samples. The choice of diphone as embedding unit relied on the used data type in the model training, as we will explain later on.

The developed “encoder” neural network is a supervised learning model that takes the sounds acoustic features as input to generate their embeddings, that are connected to the sound’s linguistic features (figure 3.3). we followed the next steps to build such model:

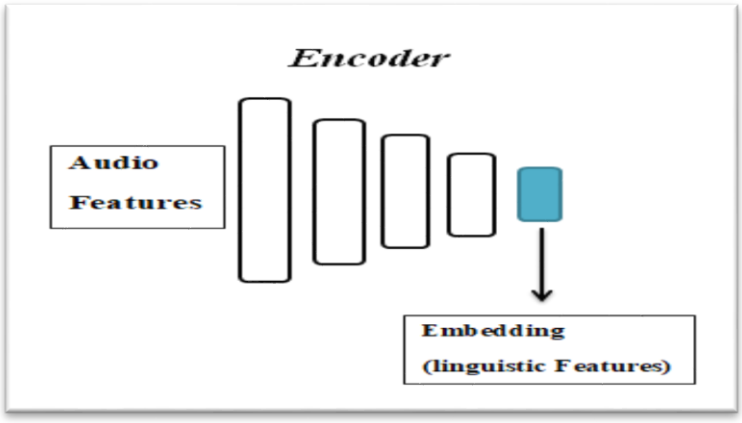


Figure 3. 3: Encoder architecture

4.1. Training and test database

To train our model we prepared for each diphone two types of datasets, as, figure 3.4shows:

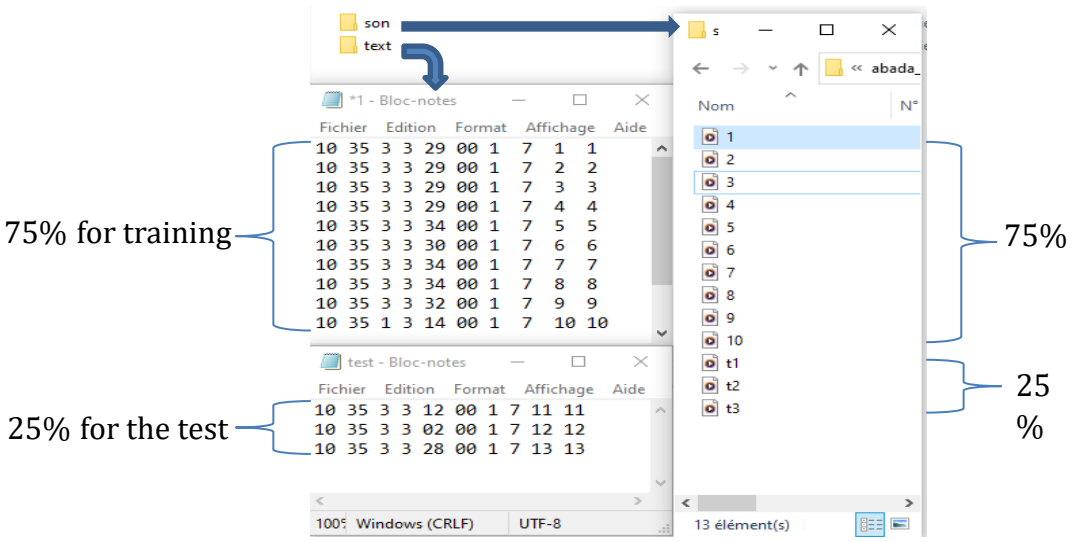


Figure 3.4: Example of data file for the diphone (r_#)

- ✓ The input dataset: consists of 100 audio files with some of their corresponding acoustic features that were extracted using Librosa library. Those sound samples were taken from the Arabic diphone database created by BETTAYEB.N [37].
- ✓ The output dataset: consists of text files that were prepared manually. These files contain 9 linguistic characteristics correspond to each sound in the input dataset. Figure 3.5, display an example of these text files, while table 3.1, present the used linguistic features.

Figure 3. 5: a linguistic text file.

Table 3. 1: the used linguistic features in our dataset

features number	Indication
1	First letter of di-phone
2	Second letter of di-phone
3	The position of the word in the sentence.
4	The position of the unit part in the word.
5 and 6	A code number indicate the left and right phonemes of the unit respectively.
7	Voiced / unvoiced.
8	Phonem type (Semi-vowels / Nasal / fricative/ Affricate/ Vibrant /Plosive).
9	Element number of di-phone.
10	The voice number in the database

Those features were coded numerically because Nympy Python library works with numbers only. In the 1st , 2nd , 5th and 6th phonemes are coded from1 to 28 according to their order in the Arabic alphabet, as explained in the first Chapter (table 1.1), the silent phoneme is coded as 35 and the vowels code is presented in table 3.2.

Table 3. 2: the vowel encoded.

ـَ : [a]	29
ـَا : [aa]	30
ـُ : [u]	31
ـُو : [uu]	32
ـِ : [i]	33
ـِي : [ii]	34

Tables : 3.3 and 3.4 P presents the chosen codes for the 7 th and 8 th feature, respectively.

Table 3. 3: the codes of the 7 th linguistic feature

feature	code
Voiced	1
Unvoiced	2

Table 3. 4: the codes of the 8 th linguistic feature

Properties of column 8	representation in the database
Semi-vowels	3
Nasal	4
Fricative	5
Affricate	6
Vibrant	7
Plosive	8
Liquid	9

4.2. The neural network architecture

In general, an encoder neural network model has a bottle neck architecture. It means that the first layer has the biggest number of neurons. Then it decreases with each layer until the last one with the smallest number of neurons, that's what make an embedding.

Our model is composed of three layers as follows: (figure 3.6)

The first layer in an LSTM with 32 neurons. We use LSTM to processing sequential data and retrain internal state.

The second and third layers are Dense type with 16 and 10 neurons respectively. We use the Dense layer in the model to add a fully connected layer. The Dense layer connects each unit in the previous layer to every unit in the current layer and has weights and biases for each connection. By using the specified activation function (in this case, **'relu'** in the first layer and **'softmax'** in the second layer). We used Denes layers with 16 then 10 neurons because it seems to us, they are the right values for our network because the data is sample and improve the model's ability to learn.

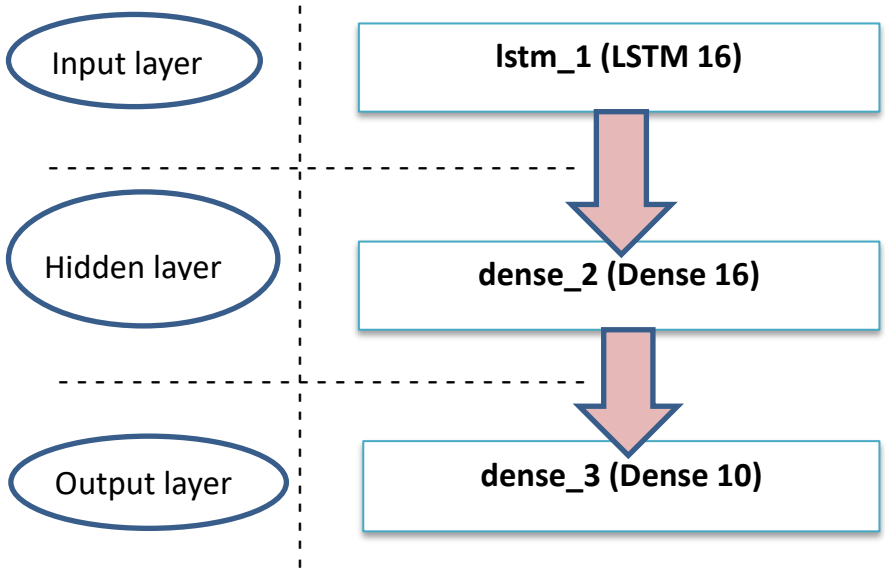
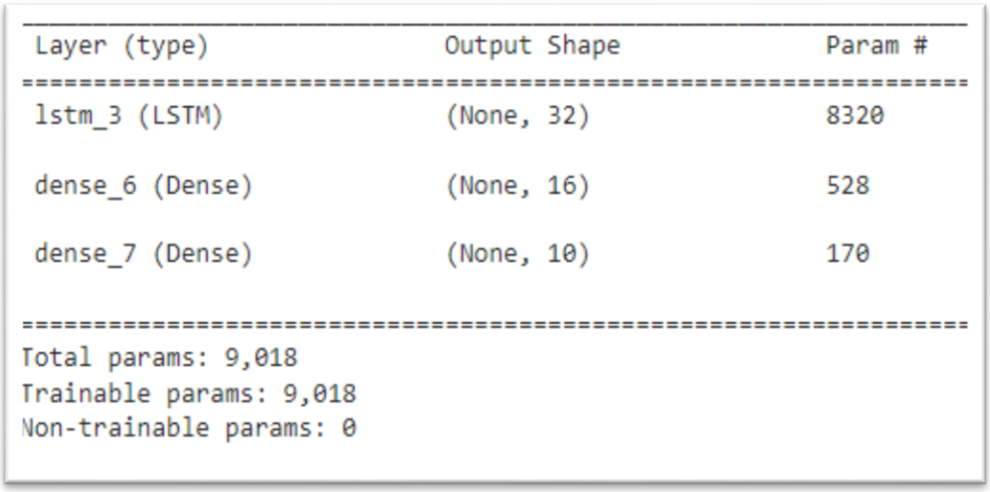


Figure 3. 6: The adopted Neural Network architecture.



Layer (type)	Output Shape	Param #
lstm_3 (LSTM)	(None, 32)	8320
dense_6 (Dense)	(None, 16)	528
dense_7 (Dense)	(None, 10)	170

Total params: 9,018
Trainable params: 9,018
Non-trainable params: 0

Figure 3.7 : characteristics of the trained neural network

4.3. The embedding building program

The overall objective of this program is to create a model that connects the sounds acoustic and linguistic features (the embeddings). This is achieved by training the model using audio samples and theirs corresponding linguistic data. After, use that model to predict the corresponding textual features for new audio data.

Figure 3.8, presents the block diagram of the program first part, that consists in:

- First, it uploads the necessary libraries like “TensorFlow”, “Numpy”, “Keras” and “Librosa”.
- Then it uploads the audio files and extract their acoustic features using “Librosa” library. The sound samples were read with a sampling rate of 44100Hz, and the extracted acoustic features are represented by the 12 MFCCs values of the sound.
- After uploading all the sound files and calculating their MFCCs values, they will be aggregated into one *{n, m,12}* dimensional array

  In witch :

  - n : represents samples’ number of the input data;
  - m : is the time serie size, represented as the maximum size of a sound file;
  - 12 : represents the number of audio features (the MFCCs)

  Generaly, the audio samples has not the same length, so they should zerro-padded into “*m”* before they were assembled togather.

- Next , the program apload the text files, containing the liunguistic features, and put their content togather in *{n ; 10}* dimentional array.
- After that, the neural network is build, as explained in the prevoiws point (4.2). then finaly it is trained with the cross entropy loss function.

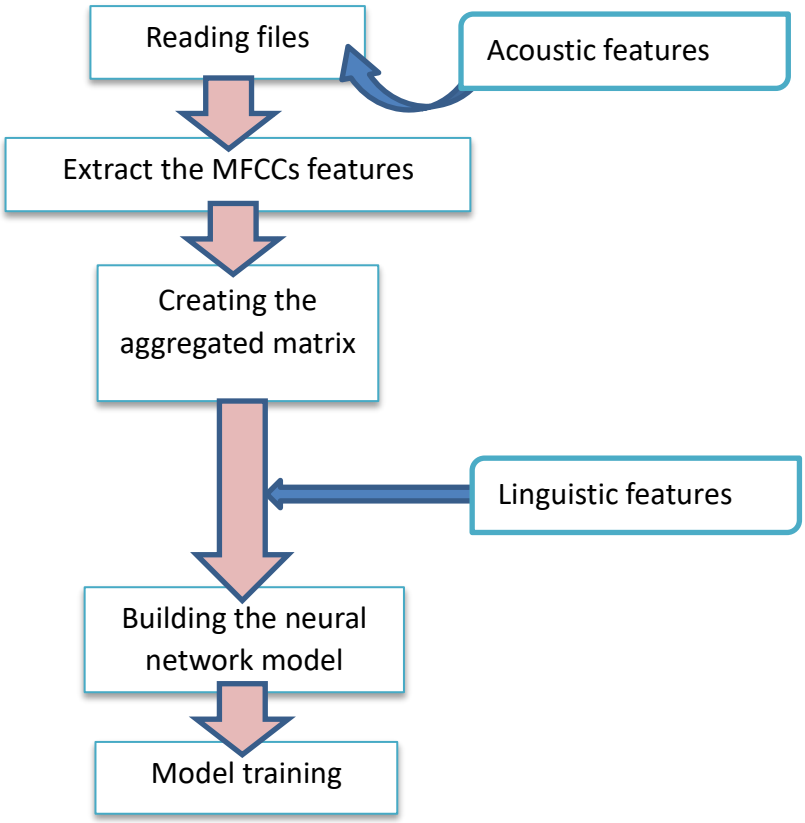


Figure 3. 8: Block diagram of the program initialization and training part.

The second part of the program is the test part. In which the trained model is given a new diphone sound and it predicts its embedded linguistic features, as shows figure 3.9.

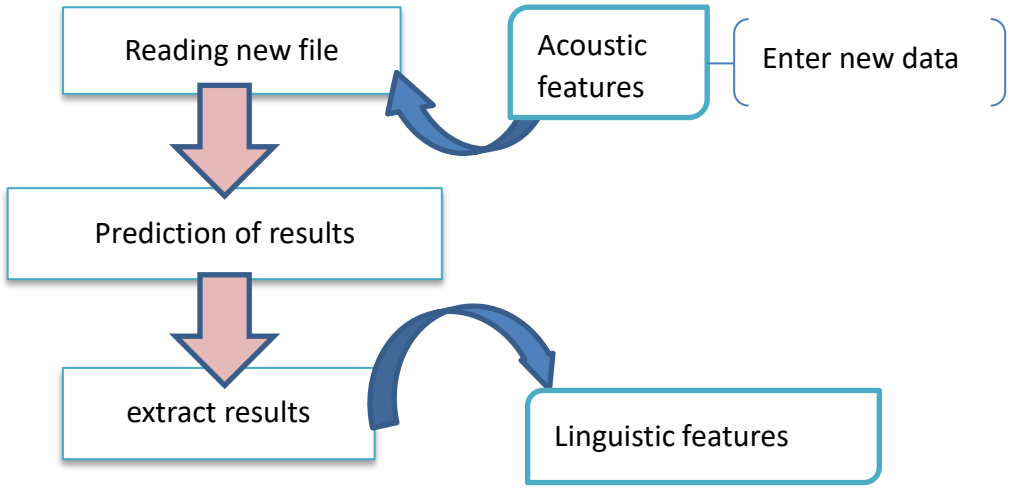


Figure 3. 9: Block diagram of the program test part.

5. Model evaluation and obtained results

5.1. Training results

After building then training the proposed model, finally, we test it. As figure 3.10 shows, we chose to divide our datasets into 75% of it for training and 25% for test.

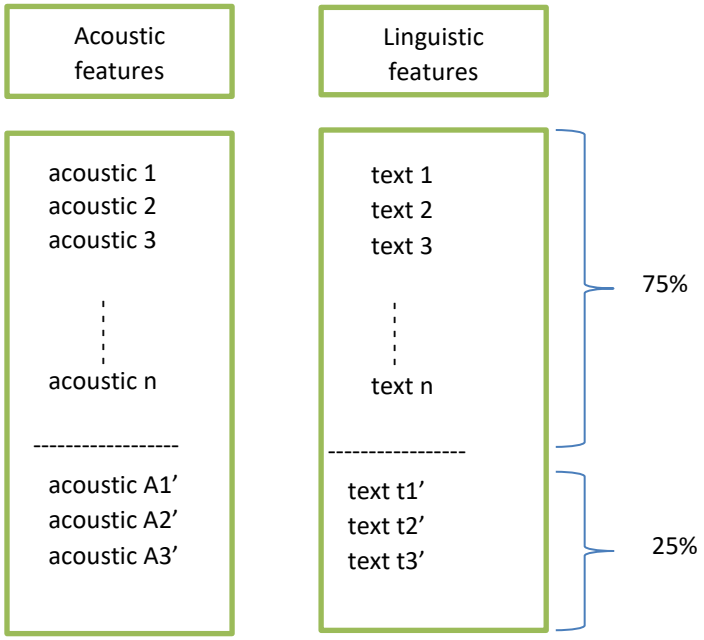


Figure 3.10: Datasets division

The training results are depicted in figure 3.11 and 3.12.

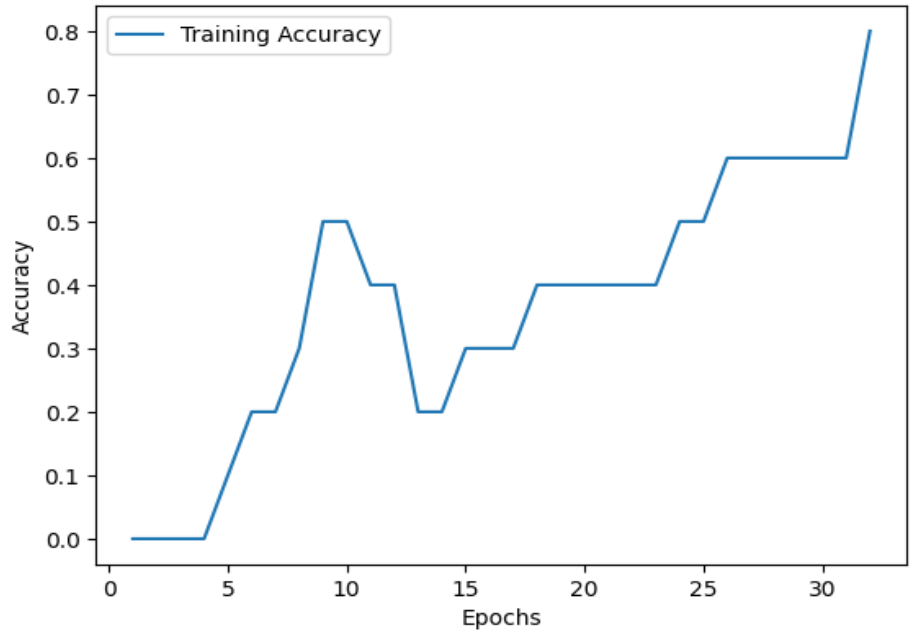


Figure 3.11: Training accuracy of the built model.

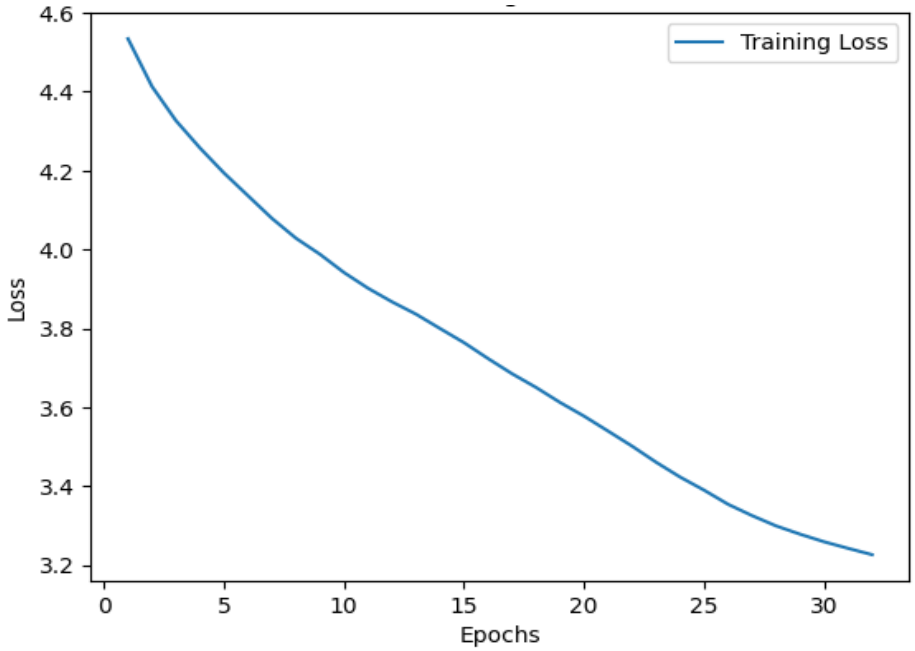


Figure 3.12: Training loss of the built model.

We observe that the training proportion increases as the number of epochs do, until it reaches a value of 80%. And the error rate decreases until it reaches 10%. This indicates that the training process was conducted effectively, as the increasing training proportion and decreasing error rate suggest a well-trained model.

5.2. Evaluation of the diphone embedding

In this part we tested our model with new data, in which we compare the predicted diphone embedding values with the linguistic features of the tested audio.

- In the first comparison, we test a di-phone sound with same characteristics as the ones used for training. We put that sound as our trained model input and we have got the results shown in figures 3.13

```
1/1 [==============================] - 1s 816ms/step - loss: 3.2123 - accuracy: 0.8000
20.00 :نسبة الخطأ بعد التدريب%
1/1 [==============================] - 1s 1s/step
linguistic_features = [ 1. 13.  2.  2. 31. 33. 12. 84.  1.  6.]
predictions = [0.01 0.128 0.02 0.02 0.31 0.33 0.12 0.42 0.01 0.06 ]
```

Figure 3.13: Results of testing mirrored features sounds.

When we compare the original linguistic features (the linguistic features of the sound that was tested) to the predicted ones , we observe that the results align with each other by approximately 80%. As when we multiply the predicted values by 100, we get the following matrix:

Predictions = [1 12.8 2 2 31 33 12 42 1 6]

Table 3.5, resumes this comparison results.

Table 3.5: Comparison results between target and predicted linguistic features

Original (target) linguistic features	[1 13 2 2 31 33 12 84 1 6]
Test results	[1 12.8 2 2 31 33 12 42 1 6]
Rounding results to decimal values	[1 13 2 2 31 33 12 42 1 6]

We consider these results acceptable because the error percentage was not that high. In general, the results were good because the linguistic features of the tested sound are very close (almost the same) to the ones used for training.

- In the second comparison, we test a diphone sound that shares some characteristics with the ones used for training. We put that sound as our trained model input and we have got the results shown in figure 3.14.

```
1/1 [==============================] - 0s 314ms/step
linguistic_features = [ 6.  5.  3.  2. 29.  0.  1.  7.  4.  4.]
predictions = [0.06 0.045 0.03 0.02 0.29 0.1 0.01 0.42 0.01 0.04 ]
```

Figure 3.14: Results of testing quite similar features sounds.

After the same process as the first comparison. we observe that the results align with each other by approximately 60%. As when we multiply the predicted values by 100, we get the following matrix

Prediction = [6 4.5 3 2 29 10 1 4.2 1 4]

Table 3.6, resumes this comparison results.

Table 3.6: Comparison results between target and predicted linguistic features

Original (target) linguistic features	[6	5	3	2	29	00	1	7	4	4]
Test results	[6	4.5	3	2	29	10	1	4.2	1	4]
Rounding results to decimal values	[6	5	3	2	29	10	1	4	1	4]

We consider these results quite good because the error rate does not exceed 50%. This is fine because the linguistic features of the tested audio are a bit deferent to the trained ones.

- In the Third comparison, we tested a di-phone sound whose features are not the same as the ones used for training. We put that sound as our trained model input and we have got the results shown in figure 3.15

```
1/1 [==============================] - 0s 353ms/step
linguistic_features = [ 8.  3.  1.  3. 14.  0.  1.  7.  1.  1.]
predictions = [0.08 0.027 0.03 0.02 0.29 0.1 0.01 0.42 0.01 0.042 ]
```

Figure 3.15: Results of testing different features sounds

The comparison results show that the target and predicted align with each other by 40%. When we multiply the values by 100, we get the following matrix:

Prediction = [8 2.7 3 2 29 1 1 42 1 4.2]

Table 3.7, resumes this comparison results.

Table 3.7 : Comparison results between target and predicted linguistic features

Original (target) linguistic features	[8 3 1 3 14 00 1 7 1 1]
Test results	[8 2.7 3 2 29 1 1 42 1 4.2]
Rounding results to decimal values	[8 3 3 2 29 1 1 42 1 4]

We got these results because the linguistic features of the tested sound are completely different from the ones on which the model was trained.

6. Conclusion

In this Chapter, we presented and explained how we built an encoder neural network model for di-phone embedding. The evaluation and tests made for that model gave encouraging results.

General conclusion

We aimed with work to develop a speech synthesis system using neural network technique. As mentioned in the first chapter this system can be used to assist people with disabilities, for educational tools or many other applications.

The work presented in this thesis consists of training a neural network model to generate an embedded representation of input sound samples. This model is the first part of an autoencoder (the encoder), and basing on it, we can build the second part that can generates the audio sounds from the embedding.

To build such model, we first, prepared our database consisting of Arabic diphone sounds and some of their corresponding acoustic and linguistic features. Then we trained the encoder with the diphones acoustic features as input to generate their linguistic features as embeddings. The model is constructed using the LSTM (Long Short-Term Memory) networks. As it they are known for their ability to capture long-term dependencies and temporal duration.

We used Python as programming language because it has a powerful ability to handle various tasks, in addition to its extensive and diverse libraries.

The obtained results from after training and testing the model were acceptable and good. As if the trained audio features and the tested audio features are similar, where the model prediction rate reached 80%. then the greater the difference between them, the model does not predict well the tested audio, and the percentage decreases.

We envision further development of the model by providing it with a larger training dataset and implementing other network architectures. This would enhance the accuracy of the text-to-speech conversion later on.

References

References

[1] : C. Touzet ,''Les Reseaux de Neurons Artificial , Introduction Au Connexionnisme'' , HAL open science, EC2, Collection de l'EERIE,2016.

[2]: B. Zhang, ''Artificial Neural Networks'', Report, School of Computer Science and Engineering, Seoul National University,2001 .

[3]: R.J. Rumelhart, "Explorations in the microstructure of cognition : Learning Internal Representations by Error Propagation", vol. 1. MIT Press, Cambridge, MA, USA. pp. 318–362 (1986). URL: http://dl.acm. org/citation.cfm?id=104279.104293.

[4]: S. Agatonovic-Kustrin , R. Beresford, "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research'', Pharmaceutical and Biomedical Analysis, August 1999.

[6]: J. Zhou , G. Cui , and others, ''Graph neural networks: A review of methods and applications '', AI open, 2020, vol. 1, p. 57-81..

[7]: E. Todt, B.Alexandre , ''Convolutional Neural Network – CNN'', Report, VRI Group - Vision Robotic and Images Federal, University of Parana´, 2019.

[8]: A. Mikami,'' Long Short-Term Memory Recurrent Neural Network Architectures for Generating Music and Japanese Lyrics'', PHD Thesis, Computer Science Department, Boston College , 2016.

[9]: D. Bank, N. Koenigstein and others,'' Autoencoders'', Report, School of Electrical Engineering, Tel Aviv University, 2021.

[10]: K. Anil Jaine, J. Mao, and others ,'Artificial neural network: A tatural', master thesis , Michigan state university, 2011.

[11]: Y. Ning and others,'' A Review of Deep Learning Based Speech Synthesis'', PHD. thesis , Research Institute of Information Technology Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China, 2019.

[12]: A.Chentir , ''Etude de la Microprosodie en vue de la Synthèse de la parole en Arabe Standard'', PHD thesis, Département d'Electronique, Ecole nationale polytechnique , Alger, 2009.

[13] : S. Baloul,' Développement d'un système automatique de synthese de la parole à partir de texte arabe standard voyelle', doctorat ,académie santé ,le Mans France,2003.

References

[14] : J.C. Wells , 'transcription and analysis', master thesis , University college London. transcription –ELL.doc .

[15]: M. Khraish, ''Arabic Phonetic Alphapet'', ninth united nations conference on the standarisation of geographical, New York, USA,2007.

[16]: E. Sicard, A.Menin-Sicard,' ''Le spectrogramme et son application en clinique orthophonique'', Report, Laboratoire LURCO, 2021, ffhal-03107434f

[17] : B. Nadjla,' Elaboration d'un Système de Synthèse par Sélection d'Unités en Vue de la Récitation du Saint Coran', PHD thesis, Ecole Nationale Polytechnique, Alger, 2021.

[18] : https://www.asha.org/public/speech/development/speech-and-language/#:~:text=Speech last visited on :23/02/2023

[19] : https://faculty.ksu.edu.sa/ar/nsaldayel/course-material/55265 last visited on 27/02/2023

[20] : https://learning.aljazeera.net/ar/blogs/pages/21888#:~:text last visited on 27/02/2023

[21] : https://learning.aljazeera.net/ar/blogs/pages/21888#:~:text last visited on 03/03/2023

[22] : https://www.cairn.info/la-phonetique--9782130653356-page-58.htm last visited on 03/03/2023

[23] : J.VANDERPLAS. "Python data science handbook: Essential tools for working with data:. " O'Reilly Media, Inc., 2016, ISBN : 978-1491912058.

[24] :https://realpython.com/ last visited on 06/05/2023

[25]: Available on Amazon: https://www.amazon.com/NumPy-Beginners-Ivan-Idris/dp/1783981967 last visited on 05/05/2023

[26]: TensorFlow documents, : https://www.tensorflow.org/api_docs last visited on 05/05/2023 23:05.

[27]: https://www.datacamp.com/community/tutorials/deep-learning-python last visited on 08/05/2023 23:26 ,

[28]: https://librosa.org last visited on _03/05/ 23:00

[29]: B.Nadjla , M. Guerti, "A Study to Build a Holy Quran Text-To-Speech System". International Journal on Islamic Applications in Computer Science And Technology, 2019 7(4), pp :1-10

References

[30]: https://www.researchgate.net/figure/A-diagram-of-the-human-vocal-production-apparatus_fig1_277131520.

[31]: https://www.researchgate.net/figure/Extraction-Mel-frequency-cepstral-coefficients-MFCC-from-the-audio-recording-signals_fig1_289375827.

[32]: https://www.almrsal.com/post/810163 .