burkina institute of technology
Educating a New Generation of Leaders

Ministry of Higher Education,
Research and Innovation

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE

Speciality: Computer Science and Entrepreneurship

Voice Translation Mobile App using Artificial Intelligence: case of Mooré and English

Presented and publicly defended by:

KONGO Hamado

Thesis' ID: .....

Supervised by: Dr. Justin KOURAOGO
Assistant Professor,
Researcher in the IT department
at UFR SEA of UJKZ

President of the jury: Dr. Herman YANOGO
Assistant Professor

Under the supervision of: Mrs. SOUDRE Flora Doris Wendnemi
Master 2, BIT

Member of the jury: M. Mahamadou KONATE

Academic year: 2023 – 2024

Dedication

I dedicate this work to my mother **COMPAORE Asséto** for giving me the chance to see this world, for supporting me during good and bad moments of my life and for educating me to become this kind of young man. Without her advice, moral and financial support, I would not have been able to be at BIT nor do this thesis. She also inspires me to do this work as she did not go to school and she always use to tell us, me and my brothers and sisters that she wanted to be able to be understood by those that went to school but she could not.

Acknowledgements

I would like to extend my heartfelt acknowledgements to:

- Susanne PERTL for having the great idea of founding BIT.
- Our Dean, Prof. François ZOUGMORE for helping me with his recommendation for my data collection.
- My thesis director Dr. KOURAOGO Justin and my thesis co-director Mrs. SOUDRE Flora for their guidance and support during this thesis writing.
- The administration and the pupils of Schorge and Saint Joseph Moukassa that helped me during my data collection phase.
- The students of BIT, UNZ, UJKZ, UIT for helping me during my data collection phase.
- The staff of CITADEL for helping me with some documents dealing with my topic.
- All our teachers from the first year to the last year that provide us invaluable knowledge in Computer Science and Entrepreneurship and specially Mr. Go Issa TRAORE, our Machine Learning Teacher for helping me to understand more about NLP.
- All the BIT Staff that support me during this adventure.
- All my family members and friends who supported me during this thesis writing process.

I can’t name everyone, but I would like to thank you all for your help during this work.

You are all unforgettable !

Preface

Burkina Institute of Technology (BIT) was inaugurated on October 15, 2018, in Koudougou. It is a private institute founded by a German woman named Susanne PERTL. The vision of BIT is to educate new generation of Leaders. The students are trained with technical and entrepreneurial knowledges to help them to achieve this goal. That is also why the classes at BIT are in English to make easier the internationalization of their students.

BIT offers classes in three majors for the Bachelor Degree and one major for Master Degree.

The majors for Bachelor's Degree are:

- Computer Science: this was the first major when the institute was inaugurated. There are courses in web, mobile, and desktop development, basics in networks, artificial intelligence and data analysis.
- Electrical Engineering: this was the second major opened in 2019 with the specialization in renewable energies. In this field, all the required courses related to renewable energies in the Electrical Engineering domain are provided.
- Mechanical Engineering: BIT added it in 2021 with a possibility to do a specialization in mining or agriculture during the last year.

BIT has a laboratory where students are doing some practical experiences. Let us specify that all these trainings are completed in three (3) years and open to everyone that has a scientific baccalaureate diploma.

The major in Master Degree, added in 2023 is about artificial intelligence. It gives strong skills in AI to allow the students to master this field of the future.

At the end of the 3 years, a thesis defense is required to obtain the bachelor degree. It is in this context that we undertook the realization of this project.

Abstract

This study addresses communication challenges between unschooled traders and their business partners by proposing a Mooré-English voice translation mobile app using Artificial Intelligence. The main goal of this project is to unleash language barrier between unschooled traders and their business partners. We collected the data ourselves, tested several algorithms and selected the best one which was a combination of CNN and Bi-LSTM.

The scientific contribution in this project can be split in two parts: the first one is the collection of basic greetings in Mooré language to contribute to reduce the data scarcity challenge of our local languages. The second one concern the Mooré speech recognition model that can help to develop other project linked to our local languages like this one.

Key Words:

- Artificial Intelligence
- Deep Learning
- Natural Language Processing
- Voice Translation
- Mobile Application

Résumé

Cette étude aborde les défis de communication entre les commerçants non scolarisés et leurs partenaires commerciaux en proposant une application mobile de traduction vocale Mooré-Anglais utilisant l'intelligence artificielle. L'objectif principal de ce projet est d'éliminer les barrières linguistiques entre les commerçants non scolarisés et leurs partenaires commerciaux. Nous avons nous-mêmes collecté les données, testé de nombreux algorithmes et nous avons sélectionné le meilleur qui était la combinaison de CNN et Bi-LSTM.

La contribution scientifique de ce projet peut être divisée en deux parties : la première est la collecte des salutations de base en langue Mooré pour contribuer à réduire le défi de la rareté de données de nos langues locales. La seconde concerne le modèle de reconnaissance de la parole en Mooré qui peut aider à développer d'autres projets liés à nos langues locales comme celui-ci.

Mots clés :

- Intelligence Artificielle
- Apprentissage profond
- Traitement du langage naturel
- Traduction vocale
- Application mobile

Table of Content

List of Acronyms and Abbreviations

AGI	Artificial General Intelligence
AI	Artificial Intelligence
ANI	Artificial Narrow Intelligence
API	Application Programming Interface
ASI	Artificial Super Intelligence
BILSTM or BI-LSTM	Bidirectional LSTM
BIT	Burkina Institute of Technology
BLEU	Bilingual Evaluation Understudy
BPM	Business Process Model
BPMN	Business Process Model and Notation
CITADEL	Centre d'Excellence Interdisciplinaire en Intelligence Artificiel pour le Developpement
CNN	Convolutional Neural Network
COCOMO	Constructive Cost Model
DFD	Data Flow Diagram
DL	Deep Learning
DNN	Deep Neural Network
DWT	Discrete Wavelet Transform
ER	Entity Relationship
FCFA	Francs des Communautés Financières d'Afrique
FFR	Fon-French
GLEU	Google-BiLingual Evaluation Understudy
GMM	Gaussian Mixture Model
GRU	Gated Recurrent Unit
HMM	Hidden Markov Model
HTTPS	Hypertext Transfer Protocol Secure

IBM	International Business Machines Corporation
IUT	Institut Universitaire de Technologie
KLSL	Kilo of Source Line Delivered
LSTM	Long Short-Term Memory
MFCC	Mel-Frequency cepstral coefficients
ML	Machine Learning
MP	Matching Pursuit
MT	Machine Translation
NLG	Natural Language Generation
NLP	Natural Language Processing
NLU	Natural Language Understanding
NN	Neural Network
OMG	Object Management Group
PHC	Population and Housing Census
RAM	Random Access Memory
REST	Representational State Transfer
RL	Reinforcement learning
RNN	Recurrent Neural Network
SOAP	Simple Object Access Protocol
SSD	Solid State Drive
SVM	Support Vector Machine
UI	User Interface
UJKZ	Université Joseph KI ZERBO
UML	Unified Modeling Language
UNICEF	United Nations International Children's Emergency Fund
UNZ	Université Norbert ZONGO
UP	Unified Process
URL	Uniform Resource Locator

USD	United State Dollar
UX	User Experience
WWII	World War II
XML	Extensible Markup Language

List of Tables

List of Figures

General Introduction

Burkina Faso is a country located in West Africa. Like the other countries in this location, a lot of Burkina Faso people don't have the chance to go to school. Based on the 5th Population and Housing Census (PHC) of 2019 in Burkina Faso [1], only 29.7% of people older than 15 years old went to school. Among them, as they can't work in public institutions, many become traders by deciding to sell traditional clothes, smartphones, crops (rice, corn, millet), traditional craft products... They work with international partners like Chinese, Arabs, French, Germans, British and Americans … These ones often face some difficulties in communication.

To overcome these difficulties, with the emergence of solutions like Siri, google translate, and Alexa that can translate speeches from one language to another one, we think about using AI technics to provide a similar solution to our local traders and their business partners. Considering that most of their international partners understand and are able to speak English, and also that Mooré is the most spoken language in Burkina Faso according to the 5th PHC with a rate of 52.9%, we were guided by the choice of the topic: "Voice Translation Mobile App using Artificial Intelligence: case of Mooré and English ".

The problematic we aim to solve through this work is: How can Artificial Intelligence help to unleash language barrier between unschooled traders and their business partners?

The implementation of this kind of solution will not only improve unschooled traders' life by making easier their communication with their partners, but it will also permit them to save more money reducing the fees of hiring a human interpreter and allowing them to avoid time losing for learning a new language.

This document is structured into two main parts: the first part is the Theoretical Studies including two chapters: Chapter 1 where we give general information related to the topic and Chapter 2 where we explore the methodologies and conceptual approaches. The second part, Design and Implementation, which covers two chapters also: Chapter 1 which deals with modelling and design while Chapter 2 delves into the project's implementation, results, and discussions.

PART I: Theoretical Studies

CHAPTER 1: General Information

Introduction:

Before moving further in our work, it is necessary to give general information about the project, describe it and give its main features. We will also talk about the context of our work, as well as the problematic and the objectives that we want to reach. This is what we are going to do in this chapter.

I. Generalities

1. Context

When we were child, as she was not sent to school, we use to hear our mother saying that she would have liked to be able to speak with those that went to school. In addition, we have participated to the defense of some alumni were the jury members asked them about how those that did not go to school can use their solutions. Then we started to think about something that we can do and that can be used by everyone, including those that did not attend school. As their main problem is the language barrier, we thought that it was necessary to find some solutions to unleash this barrier.

2. Problematic

In Burkina Faso, the adult literacy rate is 34% according to the world bank in 2022[2]. That means that, 67% of the people can't read and write. English is the most widely spoken international language. Most of the international partners of our country speak English. For example, the French people, the Chinese, the British, Americans and Arabians are able to speak English. Based on this, we can say that English is really important for everyone including people that did not go to school because it will help them to exchange easily with foreigners. Furthermore, some of the people that did not go to school are doing trading, then sometimes they have to travel for business purposes or their partners come here to exchange with them, so they need English to exchange easily with their business partners.

Sometimes:

- They spend a lot of money hiring people to translate for them, including covering transport, accommodation and food fees for them during their business trips.
- They lose time and money to learn English before travelling.

- Their VISAs are refused sometimes because they do not have English certificates.
- They cannot access to some business opportunities because they are not able to express themselves in English.

Based on the problems above, the purpose of this study is to answer the following question: How can Artificial Intelligence help to unleash language barrier between unschooled traders and their business partners?

3. Objectives

3.1 General Objective

The general objective of this study is to develop a mobile app powered by artificial intelligence to unleash language barrier between unschooled traders and their business partners.

3.2 Specific Objectives

Our specific objectives are to:

- Optimize the cost of translation: develop a cost-effective translation mobile app that reduces the need for human translators.
- Optimize the time: with this app, traders will not need to go for some months or years of English training before travelling for business purpose.
- Make easier the communication: that can be possible with the voice translation from Mooré to English and English to Mooré.
- Set up an intuitive mobile app: design a simple and intuitive user interface that will be easy to use by users with no formal education.

4. Expected results

The expected results of this project are:

- Reduced Translation Costs: traders saved a lot of money by reducing dependence on human translators.
- Optimized Time for Language Learning: traders can engage in international business and travel without needing extensive language training.
- Enhanced Communication with International Partners: improved ability for unschooled traders to communicate easily with their business partners.

- A tailored mobile app built: making easier the usage by unschooled traders.

5. Study of the existing

Since a long time, there are some studies in the world about voice recognition and voice translation using AI among which we met:

Firstly, it was **Machine Aided Voice Translation**: [3] done to help the Air Force interrogation personnel to interact with potential prisoner in an unfamiliar foreign language. In this case it was English to Spanish where the author describes the difficulties to handles each language features for the solution implementation.

Secondly**,** there is the article about **Urban Sound Classification using Long Short-Term Memory Neural Network**[4] implemented by Iurii Lezhenin and Natalia Bogach from Institute of Computer Science and Technology, Peter the Great St.Petersburg Polytechnic University and Evgeny Pyshkin from Software Engineering Lab University of Aizu propose and examine a LSTM model for urban sound classification. They used the UrbanSound8K audio dataset. The proposed network is evaluated using 5-fold cross-validation and compared with the CNN. It is shown that the LSTM model outperforms a set of existing solutions and is more accurate and confident than the CNN.

Particularly, concerning the low-resource language in Africa, there are similar works for local languages through AI that we met.

The first is **FFR V1.0 : FON-FRENCH NEURAL MACHINE TRANSLATION**[5]**,** done by Bonaventure F. P. Dossou & Chris C. Emezue who proposed a model for Fon to French machine translation based on the GRU architecture. The results were evaluated using two metrics, BLEU and GLEU. Two types of model structure were used: with diacritics and without diacritics. They showed that the model with diacritic coding gave better performance: 30.55% (BLEU) and 18.18% (GLEU) against 24.53% (BLEU) and 13.0% (GLEU) for the model without diacritical coding.

In the second place**,** we have, **NEURAL MACHINE TRANSLATION FOR SOUTH AFRICA'S OFFICIAL LANGUAGES [6]** implemented by Laura Martinus, Jason Webster, Joanne Moonsamy, Moses Shaba Jnr, Ridha Moosa and Robert Fairon. They proposed MT models to translate the official languages of South Africa into English. Their objectives were to create a BLEU reference score for neural translation between English and the ten (10) official

languages of South Africa. Their model was based on the Transformer architecture, which has a performance of point in MT. Their data source was the JW300. The results showed a BLUE score of 40%.

The third place is occupied by **English2Gbe** [7]**:** A multilingual machine translation model for Fon/EweGbe by Gilles Hacheme . He proposed a multilingual machine translation model from English into Fon and Ewe, two languages of the Gbe family. Their main objective was to show the interest of a multilingual machine translation model. At the end, they built three translation models. One model for bilingual translation from English into ewe, another bilingual model from English to Fon and a multilingual model from English into Ewe and Fon. The comparison was carried out using the BLEU metric. They used the transformers architecture. The data used for their experiment came from JW300.

As far as Burkina Faso's languages are concerned, there are some similar studies using AI already done.

The first one is the job of Faissal CONGO with the topic **Conception d'un modèle de reconnaissance vocale des langues a tons : cas du mooré** [8] where he used CNNs to build a speech recognition model for polysemous words in the Mooré language in order to help people to learn Mooré language. He got a precision of 81.818% for its model.

The second one was made by Hamed Joseph OUILY and the theme was **Proposition d'approches basées sur l'IA pour la traduction automatique des langues nationales du Burkina Faso** [9] where he propose a transformers model to build a text translation model from Mooré to French and from French to Mooré. He got a BLEU score of 65.75% for the model Mooré to French and 71.18% for the model French to Mooré.

Thirdly it is **Synthèse vocale pour les langues nationales africaines : Exemple de la langue nationale mooré (Burkina Faso)** [10] done by Maïmouna OUATTARA. She tried to build a Text to Speech model for Mooré language by using Tacotron 2 and WaveGlow. She didn't get a good result from that because the generated audio was not understandable and contain noise according to her document.

The fourth one is **Mise en place d'un modèle de reconnaissance vocale pour des langues peu dotées**[11] by Mr. BATIONO Fabrice and Mr. KEITA Zakaria Cheick Oumar. They used the combination of CNN and BiLSTM to build a voice recognition model of the Dioula numbers

1,2,3 and 4. The model had to enable the Dioula linguistic community to have access to malaria awareness messages. This was done by means of a voice interface that allow them to choose the number of the message to listen to, based on suggestions made in advance.

The last one is **Automatic pure speech files detection using MFCC+F0 features and LSTM classifier in multilingual context for low-resource languages**[12] by Mr. Go Issa TRAORE and Mr. Borlli Michel Jonas SOME. This study focuses on the creation of a model for distinguishing clean speech (speech without music, interaction or background sounds) from speech containing noise. The system takes into account Mooré, Dioula and Fulfuldé languages. The LSTM provided a more interesting accuracy than the other classifiers (97.62%).

II. Description and Features

1. Artificial Intelligence and Voice Translation

1.1 Artificial Intelligence

Artificial intelligence, or AI can be define as a technology that enables computers and machines to simulate human intelligence and problem-solving capabilities[13]. So, its main goal is to allow computers to think and act as possible as humans. AI can be used in many domains such as digital assistants, healthcare, robotics, automobile, agriculture, marketing …

The idea of "a machine that thinks" dates back to ancient Greece. But since the advent of computing machines important events and milestones in the evolution of artificial intelligence are:

- **1950**: Alan Turing publishes Computing Machinery and Intelligence. In this paper, Turing famous for breaking the German Enigma code during WWII and often referred to as the "father of computer science" asked the following question: "Can machines think?" From there, he offered a test, now famously known as the "Turing Test," where a human interrogator would try to distinguish between a computer and human text response. While this test has undergone much criticism since it was published, it remains an important part of the history of AI.
- **1956**: John McCarthy use the term "artificial intelligence" at the first-ever AI conference at Dartmouth College. Later that year, Allen Newell, J.C. Shaw, and Herbert Simon created the Logic Theorist, the first-ever running AI software program.

- **1967**: Frank Rosenblatt builds the Mark 1 Perceptron, the first computer based on a neural network that "learned" though trial and error. Just a year later, Marvin Minsky and Seymour Papert publish a book titled Perceptrons, which becomes both the landmark work on neural networks and, at least for a while, an argument against future neural network research projects.
- **1980s**: Neural networks which use a backpropagation algorithm to train itself become widely used in AI applications.
- **1995**: Stuart Russell and Peter Norvig publish Artificial Intelligence: A Modern Approach, which becomes one of the leading textbooks in the study of AI. In it, they delve into four potential goals or definitions of AI, which differentiates computer systems on the basis of rationality and thinking vs. acting.
- **1997**: IBM's Deep Blue beats world chess champion Garry Kasparov, in a chess match (and rematch).
- **2004**: John McCarthy writes a paper, What Is Artificial Intelligence? and proposes an often-cited definition of AI. It is “the science and engineering of making intelligent machines”.
- **2011**: IBM Watson beats champions Ken Jennings and Brad Rutter at Jeopardy.
- **2015**: Baidu's Minwa supercomputer uses a special kind of deep neural network called a convolutional neural network to identify and categorize images with a higher rate of accuracy than the average human.
- **2016**: DeepMind's AlphaGo program, powered by a deep neural network, beats Lee Sodol, the world champion Go player, in a five-game match. Later, Google purchased DeepMind for a reported USD 400 million.
- **2023**: A rise in large language models, or LLMs, such as ChatGPT, create an enormous change in performance of AI and its potential to drive enterprise value. With these new generative AI practices, deep-learning models can be pre-trained on vast amounts of raw, unlabeled data.

1.2 Voice Translation

Voice translation is composed by voice transcription, text translation and voice synthesis.

Voice transcription is the conversion of a speech into the corresponding text in the same language.

Text translation is the process of rendering a given text in a language to another language.

Voice synthesis is the fact to convert a specific text into the corresponding speech.

The image bellow resumes the history of speech recognition from 1950s to 2020s.

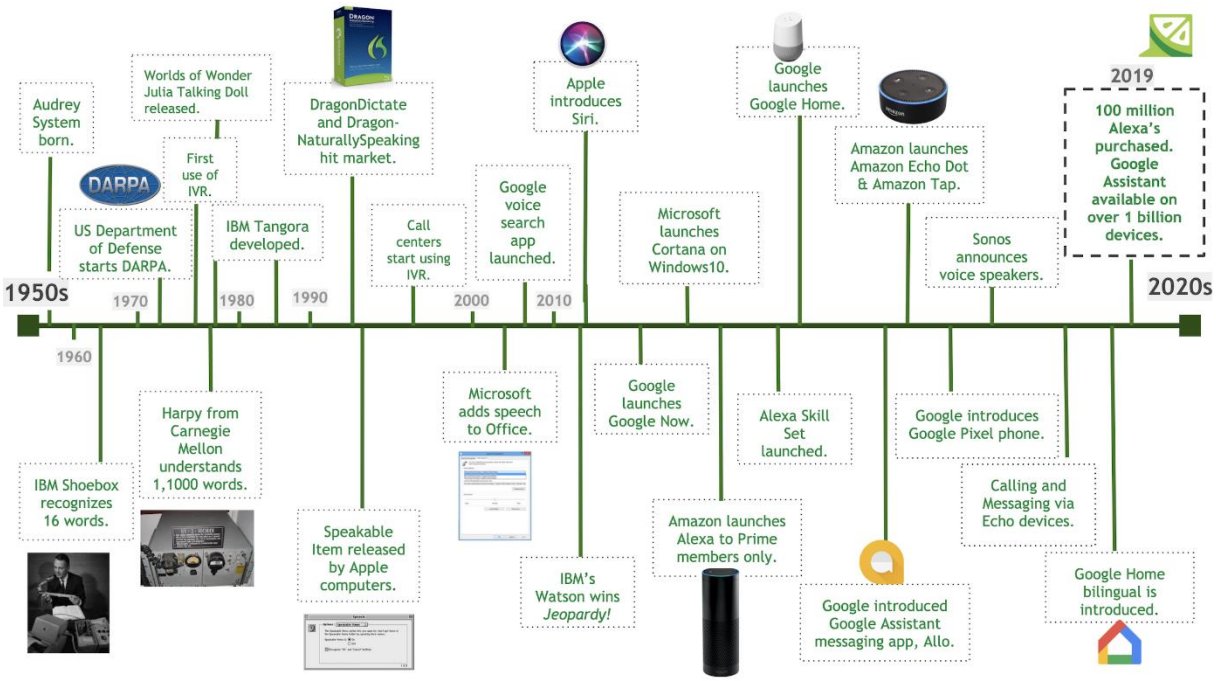

Figure 1:Speech Recognition History From 1950s to 2020s

Source: https://images.app.goo.gl/RUoVMo73ys2TDJwW9

1.3 Artificial Intelligence and Voice Translation

AI is applied in voice translation to build systems that are used to improve human life. The field of AI that deal with human voice translation is Natural Language Processing (NLP). Natural language processing (NLP) is a field of artificial intelligence focused on the interpretation and understanding of human-generated natural language. NLP techniques are employed for tasks such as Natural Language Understanding (NLU), Natural Language Generation (NLG), Machine Translation (MT), speech recognition, sentiment analysis, and more.[14]

Due to NLP, machines can learn to understand and interpret sentences or phrases, to answer questions, give advice, provide translations, and interact with humans.

2. Description of the Solution

This solution consists of using AI to build a mobile app that will be use by unschooled Mossi traders and their business partners to communicate easily by speaking in their local languages for Mossi traders and in English for the business partners in order to unleash the language barrier between them.

3. Features of the Solution

The main features of the solution are:

- ✓ **Translate voice from Mooré to English:** the trader will speak in Mooré and the app will translate it into the corresponding English speech.
- ✓ **Translate voice from English to Mooré:** the English partner will speak in English and then the app will translate it into the corresponding Mooré Speech.

Conclusion

In conclusion, this chapter gave the basis to understand the key concepts of our project, the similar projects and the importance of this kind of solution for Burkina Faso unschooled Mossi traders and their business partners. In the following lines, we will deal with methodologies and conceptual approaches.

CHAPTER 2: Methodologies and Conceptual Approaches

Introduction

After understanding the key concept of the project, in this chapter, we will explain the different methodologies and conceptual approaches that we will use to develop our solution including modeling process, modeling language, provisional schedule and the development of the system.

I. Definition

1. Methodologies

Research methodologies are the specific procedures or techniques used to identify, select, process, and analyze information about a topic. App development methodologies are approaches that provide a systematic approach to the app development process. These methodologies ensure that the final product is high quality, meets the requirements, and is delivered on time and within budget.

2. Conceptual Approaches

Conceptual approaches refer to the underlying theories and principles used to guide the understanding and interpretation of a specific problem or domain. For example, in app development, the conceptual approach might include focusing on user-centered design, ensuring that every decision we make keeps the user's needs and behaviors in mind.

II. Modeling Process

In software engineering, a software process model is the mechanism of dividing software development work into distinct phases including design, product management, and project management. It is also known as a software development life cycle.

To better understand different software process models, let's compare two methodologies: the Unified Process (UP) and the Two-Track Unified Process (2TUP). The following table highlights the key differences and similarities between them.

Table 1: Comparison between UP and 2TUP Processes

Criteria	Unified Process (UP)	Two Track Unified Process(2TUP)
Description	UP is an iterative and incremental software development framework that follows the principles of the Unified Modelling Language (UML). It provides a discipline approach to development.	2TUP is an extension of UP that focuses on concurrent development tracks: the architecture track and the solution track. It is useful for complex projects with upfront architectural analysis.
Strengths	○ Emphasizes iterative development ○ Comprehensive approach covering all development phases ○ Incorporates risk management ○ Use case-driven for requirements modelling	○ Allows concurrent development tracks ○ Provides clear separation of architecture and solution development ○ Addresses technical risks early through architecture analysis
Weaknesses	○ Can be time-consuming and complex ○ Requires strong project management and coordination ○ May not be suitable for small-scale projects with limited resources	○ Complexity may increase with multiple tracks ○ Requires upfront architectural analysis and design ○ Requires careful synchronization between architecture and solution development

Based on this comparison, the 2TUP process is a suitable choice for the present project. It offers a Y-shaped development cycle that separates the technical and functional aspects, allowing the focused and concurrent work streams.

The three branches of the 2TUP process are as follows:

- **Functional branch**: this branch concentrates on identifying functional requirements. It involves gathering requirements, understanding user needs and defining the functionalities of the solution. The results of this branch serve as the basis for developing the solution.
- **Technical branch:** this branch involves taking into account the technical requirements, the infrastructure and the architectural aspects of the system. The technical branch ensures that the necessary technologies, frameworks and infrastructure are identified and aligned with the project objectives.
- **System implementation phase**: this phase brings together the results of the functional and technical branches. It comprises several stages, including preliminary design, detailed design, coding and integration. During these stages, functional and technical aspects are combined to develop the software solution itself. The result of this phase is the operational implementation of the solution.

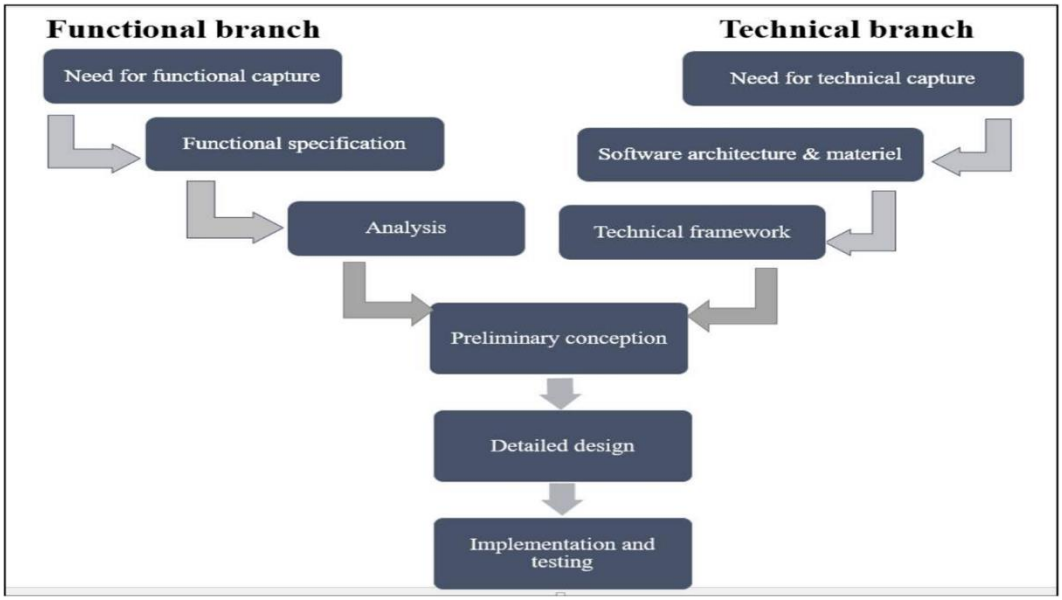

Figure 2:2TUP Branches

III. Modelling language

A modelling language is a tool that can be used to express data, information, knowledge or systems in a structure defined by a consistent set of rules. Modelling languages commonly used in software engineering and systems designing include the following: Unified Modelling Language (UML), Entity Relationship Diagrams (ER), Data Flow Diagrams (DFD), Business Process Model (BPM), Business Process Model and Notation (BPMN)...

In this project, UML will be use as a modelling language. The UML is a standardized modelling language used in software engineering to visually represent, specify, construct, and document the system. It provides a set of graphical notations for modelling various aspects of a system, including its structure, behaviour, interactions and architecture. UML is widely accepted and used in the software development industry.

The latest version of UML (UML 2.5.1) published in December 2017 by the Object Management Group (OMG) includes fourteen (14) diagram types, seven are structural and seven behavioural as shown in the figure below:

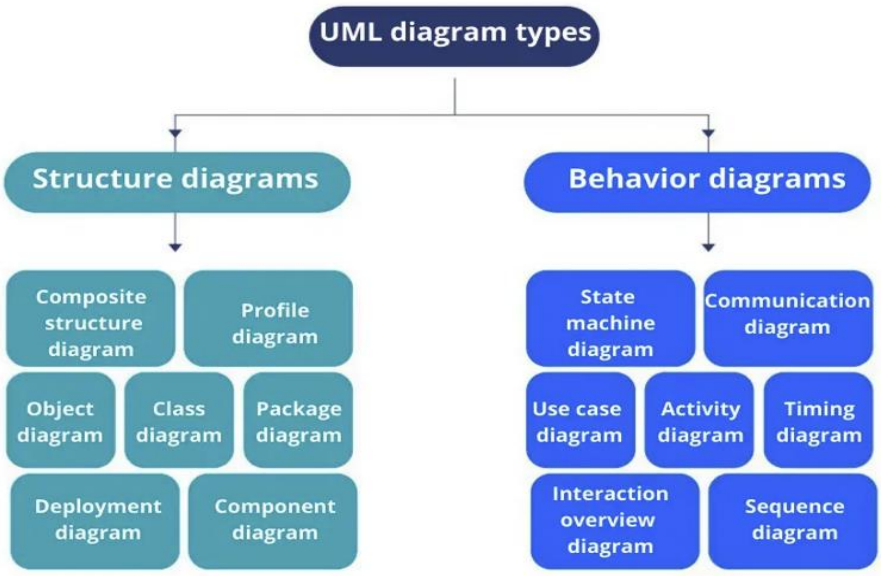

Figure 3:UML Diagram types

Source: https://images.app.goo.gl/N9NnGDKrHtWheAK46

IV. Provisional Schedule

Table 2:Provisional Schedule

Action	Start Date	Duration(days)	End date
Data collection	01/02/2024	120	31/05/2024
Literature Review	01/05/2024	50	20/06/2024
Model building	20/06/2024	30	20/07/2024
Report writing	20/06/2024	70	29/08/2024
Model testing and improvement	20/07/2024	10	30/07/2024
API building	30/07/2024	7	06/08/2024
Mobile app designing	06/08/2024	3	09/08/2024
Mobile app development	09/08/2024	7	16/08/2024
Mobile app testing and improvement	16/08/2024	3	19/08/2024
API integration in the mobile app	19/08/2024	7	26/08/2024
mobile app test and improvement	26/08/2024	3	29/08/2024

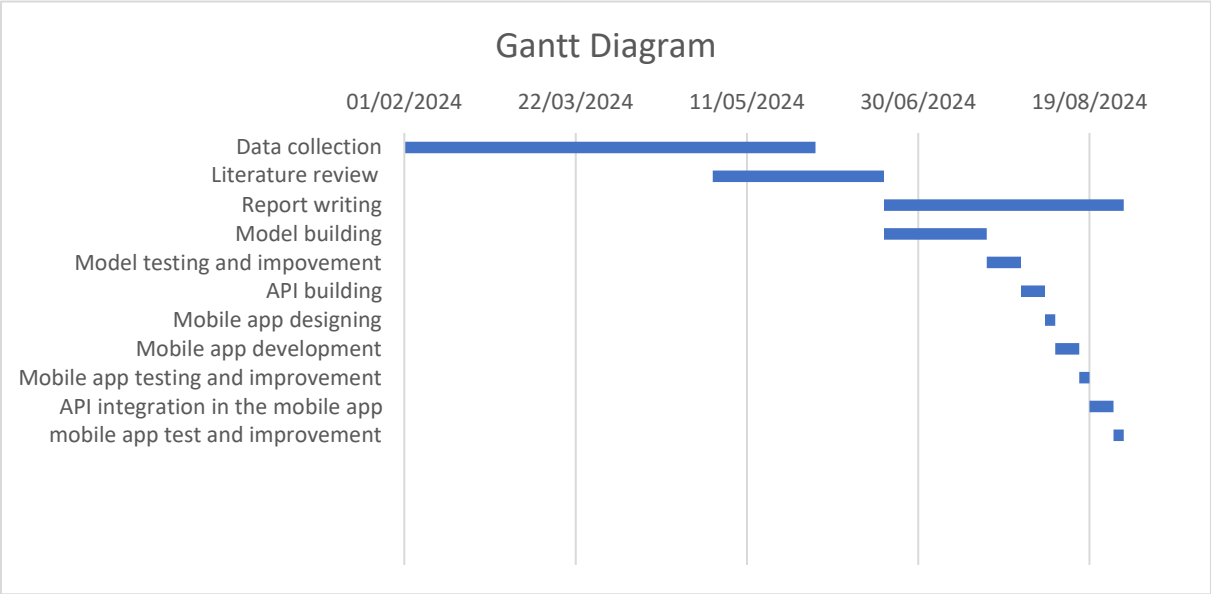

Figure 4:Gantt Diagram for provisional schedule

V. Development of the System

A programming language is a notation system for writing computer programs. Programming languages generally provide features such as type, variables and error handling mechanisms. A programming language can be either a compiler or an interpreter. An interpreter executes the source code directly, while a compiler produces an executable programme. Programming languages are mainly used to create desktop software, websites, mobile applications and commercial systems. According to the Online Historical Encyclopaedia of Programming Languages, people have created about 8,945 coding languages.

As the number of programming languages is high, we have to select the right one depending on the project (mobile, web or desktop app). Considering that our target audience are unschooled traders, we think that a mobile app suits most as most of them have smartphones instead of desktops for their business but they are not able to enter a URL in their browser to look for a web app. Based on that, we are going to choose a mobile app programming language for our solution.

1. Mobile Development

A mobile app is a software application developed specifically for use on small, wireless computing devices, such as smartphones and tablets. In the past, developers were obliged to develop a code for each mobile operating systems with android and iOS as the main ones. Nowadays developers have the possibility to use one programming language for both. This is the case of Dart programming language with its framework Flutter and JavaScript programming language with its framework React Native which are the main one used as shown in the following image.

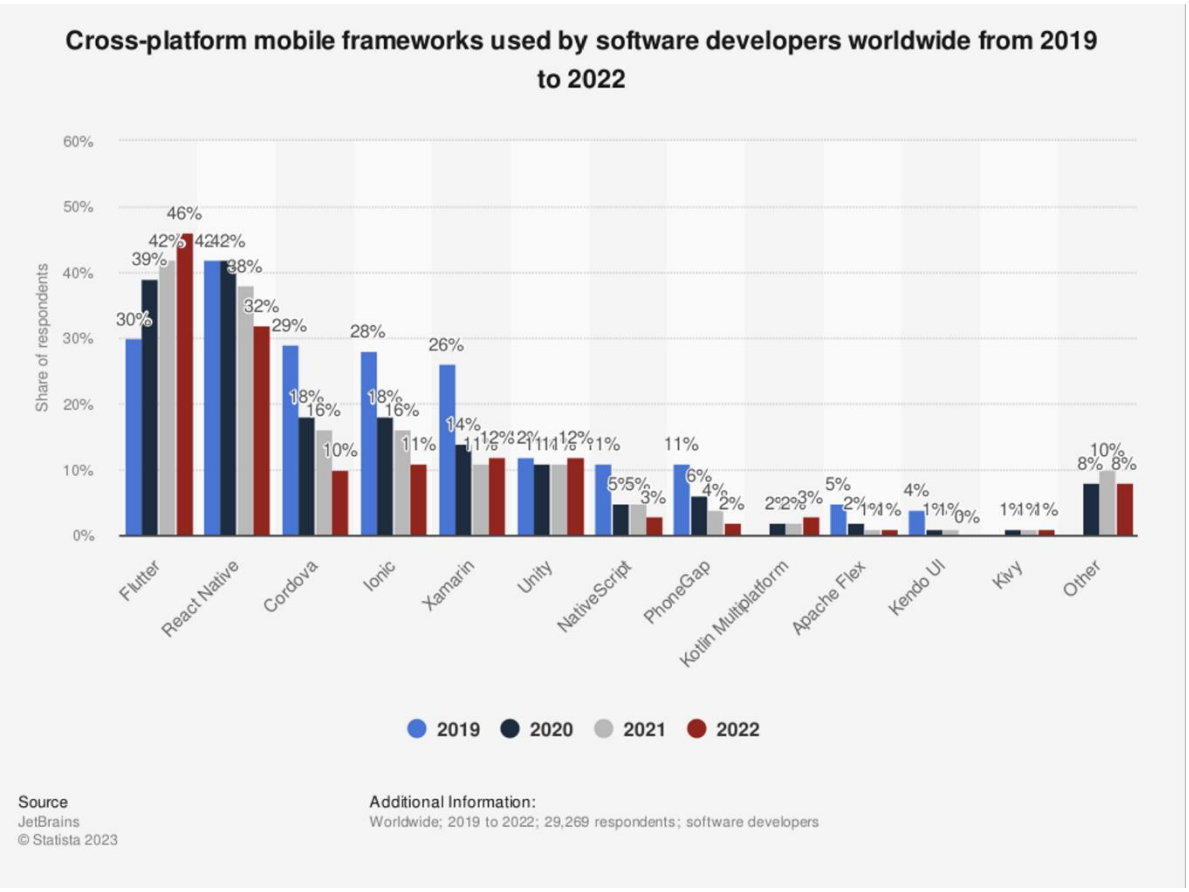

Figure 5:Number of developers per mobile frameworks from 2019 to 2022

Source: https://images.app.goo.gl/wHZEZ5KMFTJfq1q96

Let's explore now the programming languages that can be used to build AI models.

2. Artificial Intelligence's Model Development

A lot of programming languages are used to develop artificial intelligence models. We discover that the most used one is python because it has a great library ecosystem, good visualization options, a low entry barrier, a big community support, flexibility, readability, and platform independence. This is what we can figure out thanks to the following image that give us the 10 best programming languages used to build AI models. This classification was made by GeeksforGeeks.

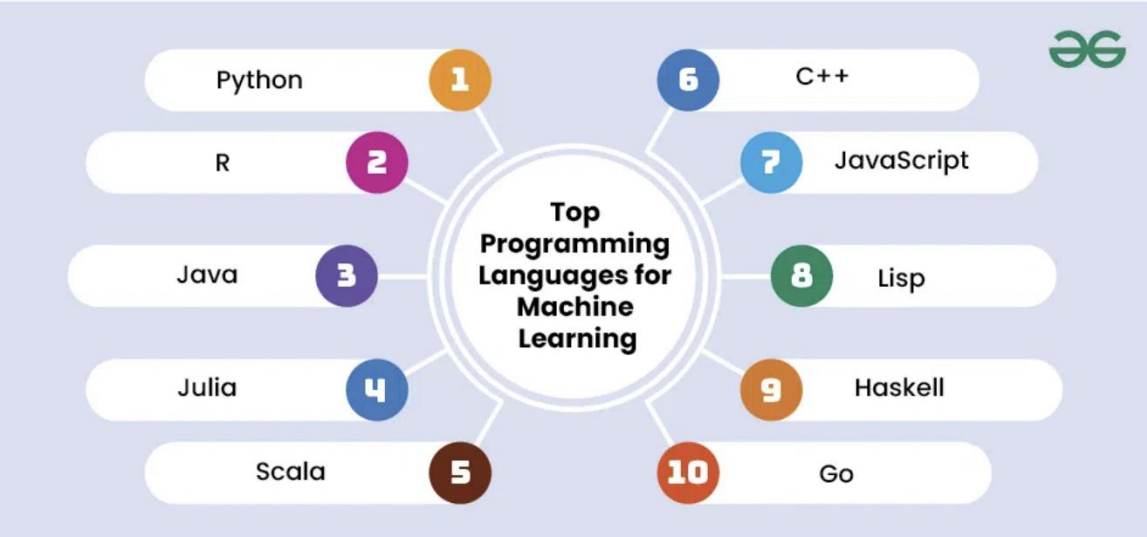

Figure 6:10 Best Languages for Machine Learning in 2024 - GeeksforGeeks

Source: https://images.app.goo.gl/mcAQCLqbadCHU15w5

After getting an overview of the main programming languages used to build AI models, let's now have a look on NLP techniques used in sound classification.

2.1 Top NLP Techniques in Sound Classification

The methods used involve k-Nearest Neighbors (k-NN) algorithm, Support Vector Machine (SVM), Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) in combination with features engineered by signal processing techniques, e.g. Mel-Frequency Cepstral Coefficients (MFCC), Discrete Wavelet Transform (DWT) coefficients and Matching Pursuit (MP) features [15]–[16]-[17]. In contrast with described approaches, deep neural networks (DNN) allow to facilitate feature engineering keeping classification accuracy and even outperform the conventional solutions [18]. In particular, being able to capture spectro-temporal patterns from spectrogram like input convolutional neural networks (CNN) have high performance [19]-[20]-[21] Long short-term memory (LSTM) networks is the other type of neural network architectures that is exploited for sound classification [22], as well as the combinations of LSTM and CNN [23], [24]. There are also BI-LSTM, the combination of CNN and BI-LSTM, and Transformers.

Let us describe more the DNNs and their combinations.

1. **CNN**: CNNs or convolutional neural networks are a type of deep learning algorithm developed by Yann LeCun in 1988 that perform well at learning images. That's because they can learn patterns that are translation invariant and have spatial hierarchies (F. Chollet, 2018). The image below resumes how CNNs work to classify images.

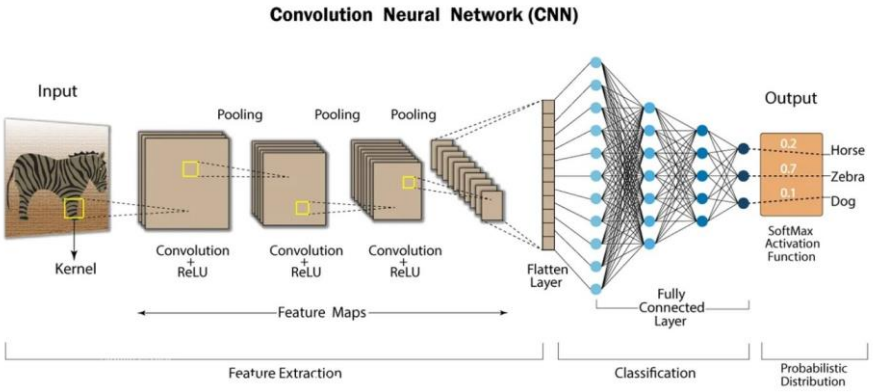

Figure 7:Example of CNN workflow

Source: https://images.app.goo.gl/muBXtoVUqrd8SRZm7

2. **LSTMs**: Long Short-Term Memory is an improved version of recurrent neural network designed by Hochreiter & Schmidhuber in 1997. A traditional RNN has a single hidden state that is passed through time, which can make it difficult for the network to learn long-term dependencies. LSTMs model solve this problem by introducing a memory cell, which is a container that can hold information for an extended period. LSTMs are capable of learning long-term dependencies in sequential data, which makes them well-suited for tasks such as language translation and speech recognition.

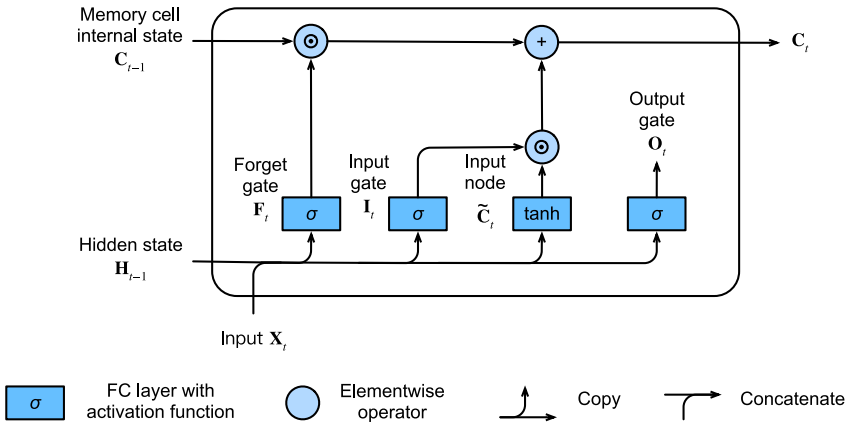

Figure 8:LSTMs' architecture

Source: https://images.app.goo.gl/VXoakvyACEQPN38R9

3. **Bidirectional LSTM (biLSTM):** It is a recurrent neural network used primarily on natural language processing designed by Graves and Schmidhuber in 2005. Unlike standard LSTM, the input flows in both directions, and it's capable of utilizing information from both sides. It's also a powerful tool for modeling the sequential dependencies between words and phrases in both directions of the sequence. BiLSTM adds one more LSTM layer, which reverses the direction of information flow. Briefly, it means that the input sequence flows backward in the additional LSTM layer.

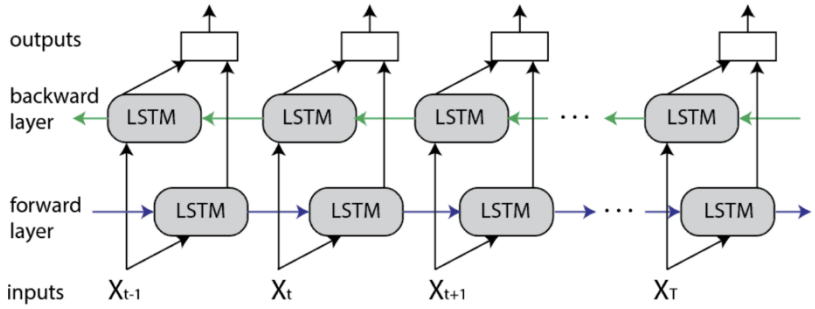

Figure 9:Bi-LSTMs' architecture

Source: https://images.app.goo.gl/hHg8ogKzEGFA6taK6

4. **Transformers**: Transformers are a deep learning architecture that relies on self-attention mechanisms to process and learn from sequential data. Unlike RNNs, they can handle long-range dependencies efficiently by dealing with all the elements of the sequence simultaneously. This makes them powerful for tasks like language translation, speech recognition, and audio classification, specially where understanding context over time is crucial. However, they require substantial computational resources and large datasets to perform optimally.

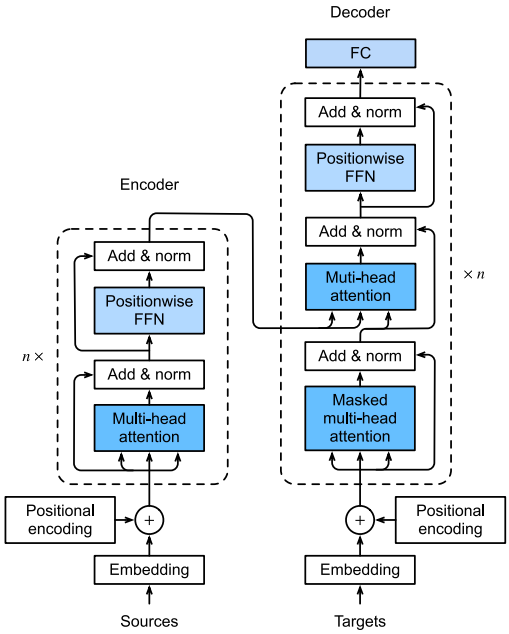

Figure 10:Transformers' architecture

Source : https://images.app.goo.gl/nEMw3Nhh1pQMyR8UA

5. **CNN and LSTM combination**: one approach for combining CNNs and LSTMs for time series classification consist to use a CNN to extract features from the time series data and then feed these features into an LSTM for classification. This allows the CNN to capture local patterns in the data while the LSTM can learn the temporal dependencies and make the final prediction.

6. **CNN and Bidirectional LSTM (biLSTM) combination:** Another approach for combining CNNs and LSTMs for time series classification consist to use a CNN to extract features from the time series data and then feed these features into an biLSTM for classification.

Based on this we will try 5 approaches (CNN, LSTM, biLSTM , Combination of CNN-LSTM, and combination of CNN-biLSTM) and select the best one. We will not try transformers approaches as they required more data and are very resource consuming than the others.

There are some frameworks that helps to build deep learning algorithms, most of them are compared in the figure bellow.

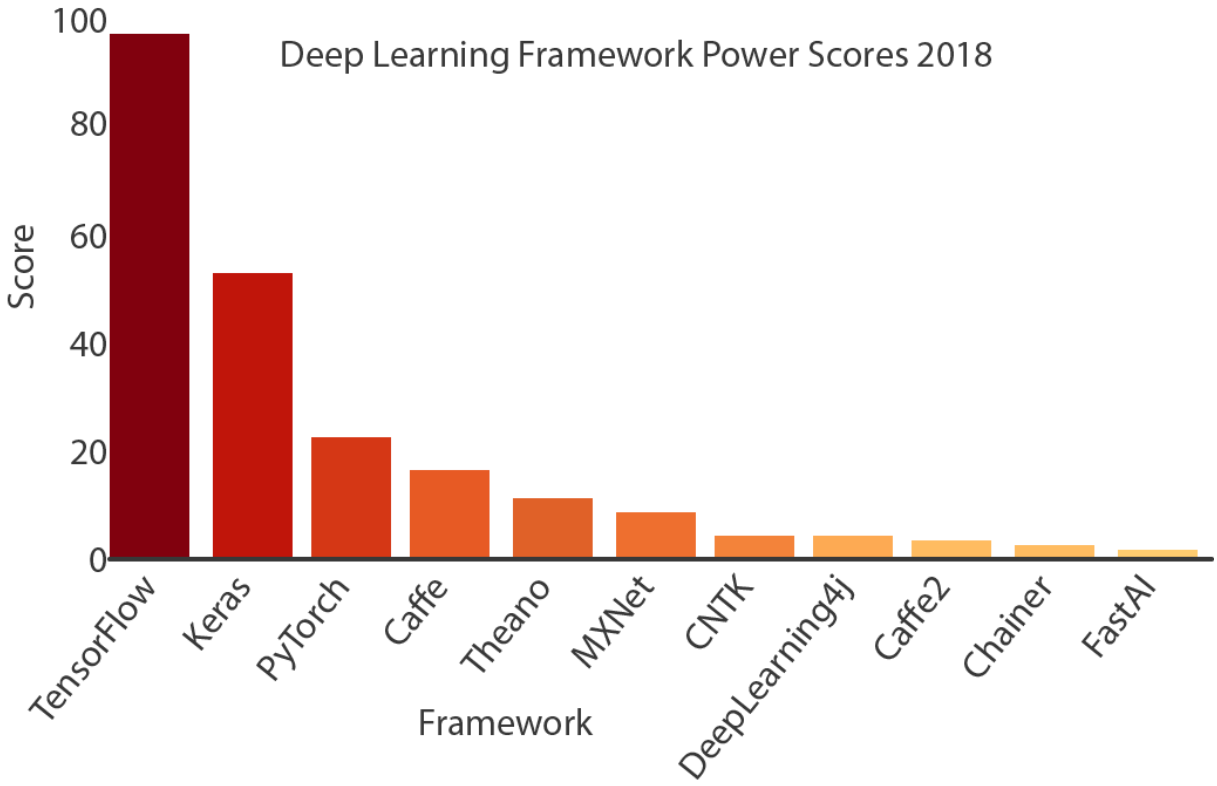

Figure 11:Deep Learning Frameworks Scores on 2018

Source : https://images.app.goo.gl/pBh4ZceJT46A9WqNA

Based on the previous image, we remark that TensorFlow is the best one followed by Keras. Let's now have a look on NLP technics used for under resourced languages.

2.2 NLP Technics for Under-resourced Languages

The term "under-resourced languages" introduced by [Krauwer, 2003; Berment, 2004] refers to a language with some of (if not all) the following aspects: lack of a unique writing system or stable orthography, limited presence on the web, lack of linguistic expertise, lack of electronic resources for speech and language processing, such as monolingual corpora, bilingual electronic dictionaries, transcribed speech data, pronunciation dictionaries, vocabulary lists, etc. Based on this definition, in Burkina Faso, our local languages are under-resourced languages because they have some characteristics like the ones above. To overcome these challenges, there are some technics used in NLP among which people use mainly:

Data augmentation: it is a technique that increases the diversity and size of the training data by applying various transformations and manipulations. For example, we can add noise, change the pitch speed, or volume, or mix different audio sources to create new synthetic samples. Data augmentation can help us to reduce overfitting, enhance the variability, and improve the performance of the speech recognition system.

Transfer Learning: it is a technique that takes advantage of the knowledge and parameters of a pre-trained model on a high-resource language and adapts them to a low-resource language. This way, we can benefit from the large amount of data and features that are already learned by the pre-trained model and fine-tune it with a smaller amount of data from the target language. Transfer learning can reduce the training time and improve the generalization and robustness of the speech recognition system.

After dealing with techniques that are used in NLP for under-resourced languages, let's now deal with a technology that will help us to serve our AI model.

3. Application Programming Interface (API) Development

Application programming interface is a software intermediary that allows two applications to talk to each other. APIs are an accessible way to extract and share data within and across organizations. SOAP and REST are two different approaches for APIs development. The SOAP approach is highly structured and uses XML data format. REST is more flexible and allows applications to exchange data in multiple formats. To implement the project, we will use REST approaches.

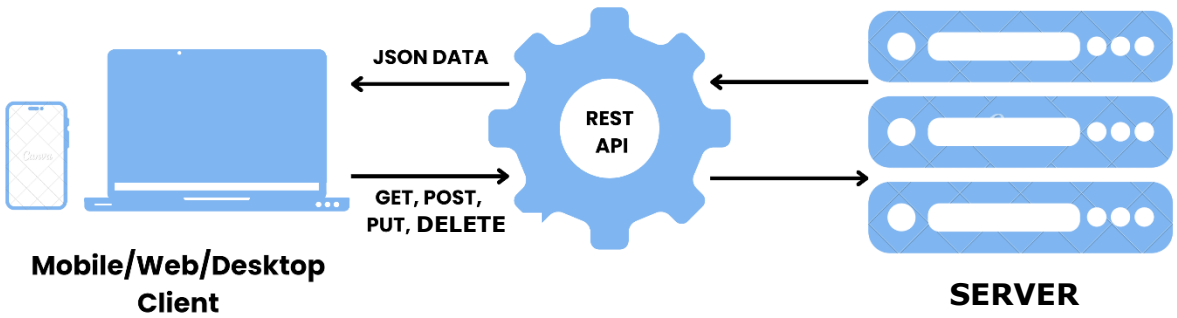

Figure 12:REST API Architecture

Advantages of using REST API in our project:

- **Data Protection:** By serving the model through an API, sensitive data can be transmitted securely using encryption protocols like HTTPS.
- **Scalability**. Systems that implement REST APIs can scale efficiently because REST optimizes client-server interactions.
- **Flexibility**: RESTful web services support total client-server separation. APIs enable remote access to the AI model from various clients.
- **Independence**. REST APIs are independent of the technology used.

After seeing the advantages of using rest APIs to serve our model, let's now see how we can build it in python through the image bellow.

	django	Flask	FastAPI
Performance speed	Normal	Faster than Django	The fastest out there
Async support	**YES** with restricted latency	**NO** needs Asyncio	**YES** native async support
Packages	**Plenty** for robust web apps	**Less than Django** for minimalistic apps	**The least of all** for building web apps faster
Popularity	The most popular	The second popular	Relatively new
Learning	Hard to learn	Easier to learn	The easiest to learn

Figure 13:Comparison between the best python frameworks for REST APIs development

Source : https://images.app.goo.gl/7KYmXkXwnq5Qkt3N6

Based on the previous image, the best python frameworks to create REST APIs are FastAPI, Flask and Django REST but FastAPI is the fastest one and the easiest one to learn among all despite the fact that it is newer than the others.

Conclusion

In conclusion, there are several methods used to develop solutions in informatics depending on what we want to achieve. In this chapter, we gave an overview about the programming languages and tools used for each specific technology that concern our project. After that, we will go now to the design and implementation phase.

PART II: DESIGN AND IMPLEMENTATION

CHAPTER 1: Modeling and design

Introduction

After finishing the theoretical part, in this chapter, we will do the preliminary study, the functional study and the technical study that will help us to implement our solution.

I. Preliminary Study

1. Preliminary Study in the Y process

At the core of the Y process, the preliminary study serves as the starting point that guides the entire project. It consists of two main phases: firstly, the clear articulation of the project's fundamental objectives and scope, and secondly, the initial identification of both functional and operational needs. These initial stages collectively provide essential direction for the development process and facilitating the creation of diagrams, which are relevant for capturing both functional and technical requirements.

2. Presentation of the Project

Our project consists to build an intuitive hybrid mobile app easy to use by unschooled traders and their business partners. The solution must allow unschooled Mossi traders to discuss easily with their foreign business partners that understand English by only speaking Mooré and the app will translate it into English for their business partner. On the other hand, the partner will speak in English and the app will translate it into Mooré for the unschooled trader.

Technically, we expect these results at the end of the project:

Fast translation: to ensure the users a fast translation after their speech as time is crucial in business.

Easy to use app: build the app based on user experience (UX) rules to allow unschooled traders to use it easily.

Accuracy: the translation must be accurate to avoid translation errors that could create problems between the two partners.

3. Initial Collection of Functional and Technical Requirements

This phase marks a turning point in the preliminary study, where the project moves from a high-level vision to concrete functionalities.

The main functionalities identified are as follows:

- **Translate a Mooré speech into English speech:** the app must translate a spoken Mooré speech into the corresponding English speech to permit to the partner of the unschooled trader to understand what he said in Mooré.
- **Translate a speech from English to Mooré:** the app must traduce a spoken English speech into the corresponding Mooré speech to allow the unschooled trader to understand what his partners said in English.

4. Functional Requirements for AI development

Here it is about listing the main requirements talking about AI development.

These are:

- **Data collection:** this part consists of collecting Mooré voices to feed our algorithm in order to allow him to recognize the Mooré speech. As we didn't found Mooré language speech datasets, we made an online form to collect it on JotForm [25]. We shared this form to our contacts and we went to some schools and universities for data collection purpose.

  We did the form to collect ten (10) basic greetings in Mooré language. You can see the form through the following image. We used a simple orthograph to make it easier to read for those submitting the form. So, it is not the correct orthograph of the Mooré language that we used in the form.

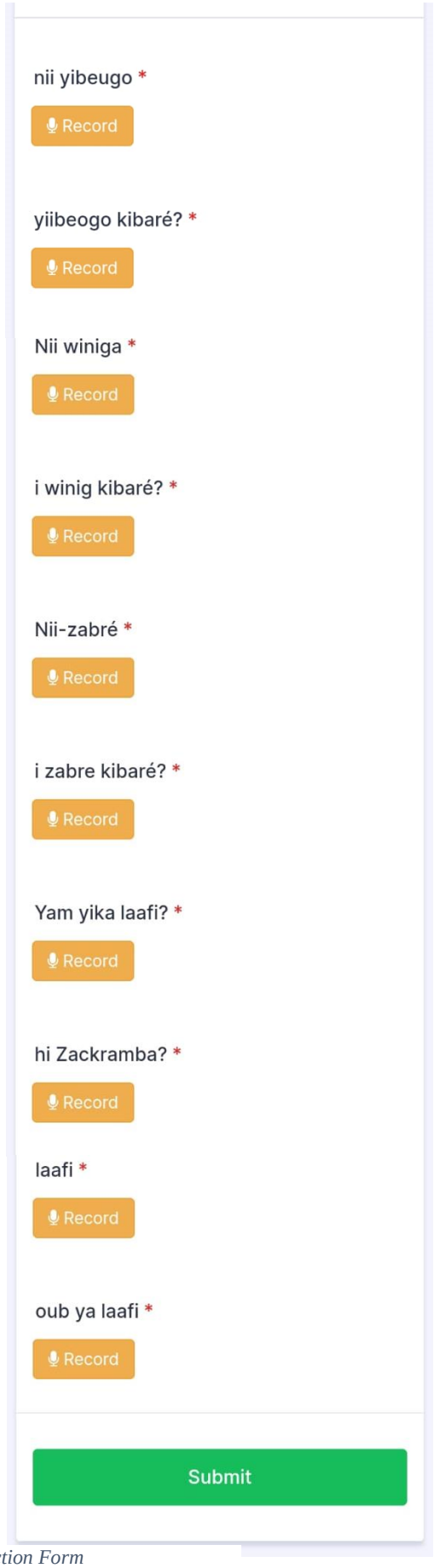

Figure 14:Data Collection Form

- **Data Analysis:** In this part we listened to each submission in order to detect and remove any bad voice record that could damage the prediction of our algorithm because we can't really trust submitters. During this process we deleted a lot of audios due to bad pronunciation, empty voice or even some discussions in French.
  At the end of this process, our dataset was represented as in the graph bellow.

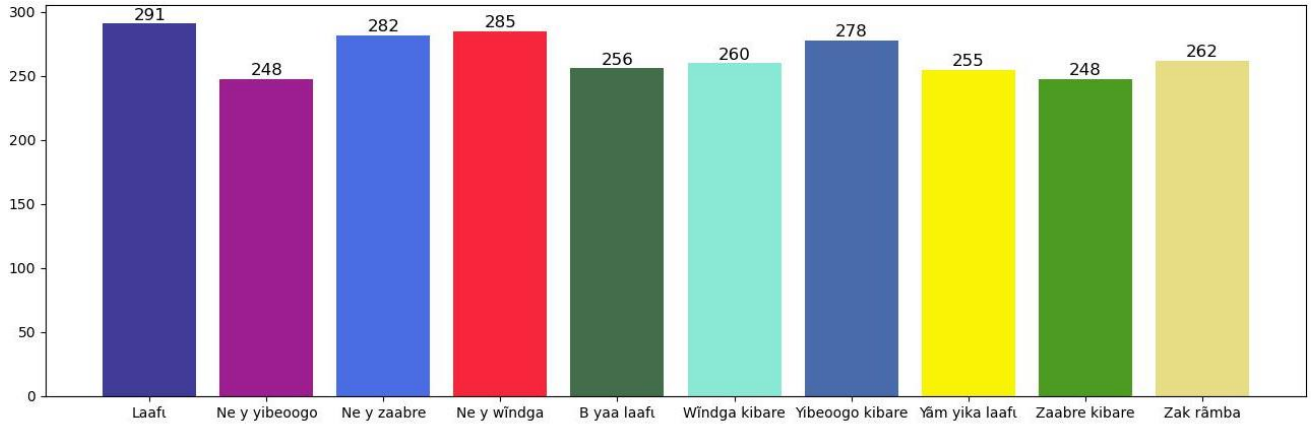

Figure 15:Initial Dataset Visualisation

You can remark that this dataset is not really sufficient to build a good DL model, then we used audio data augmentation technics to have more data. The image bellow shows our augmented dataset.

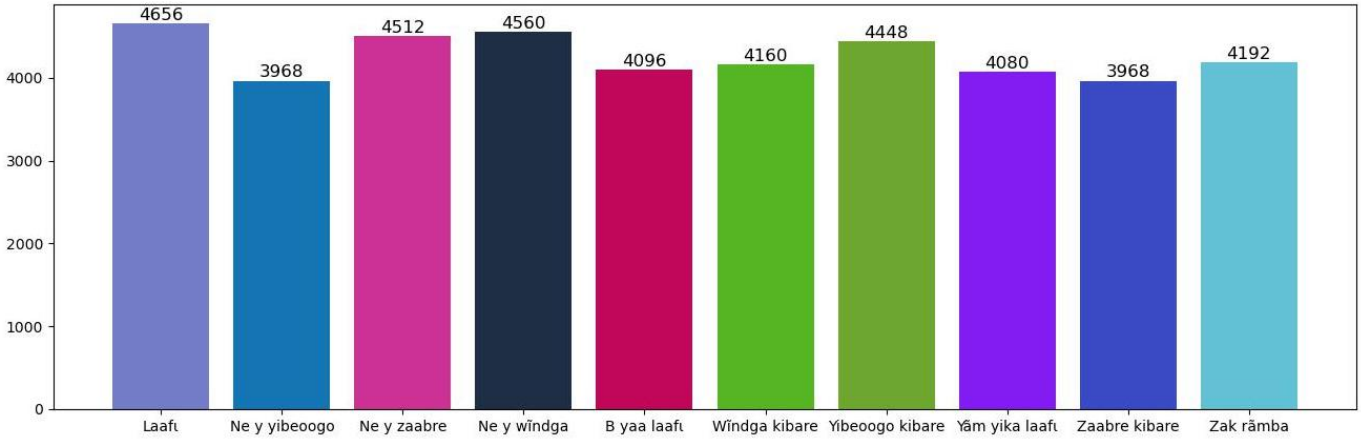

Figure 16:Augmented Dataset Visualisation

- **Development of the AI:** this section consists mainly about training our model to recognise Mooré speech.

5. Identification of the actors

The actors of our system are:

- **The unschooled trader**: this is the person that will talk to his business partner in Mooré using the app.
- **The abroad business partner** (customer, big seller): this is the person that will talk to the unschooled trader in English using the app.

II. Functional study

The functional study phase concerns a comprehensive capturing of the functional requirements of our app, followed by deep analysis and design to ensure the platform's effective functionality. This stage includes a number of phases, but the most important ones that will be highlighted in this work are: the use case diagram, the textual description, the sequence diagram and the class diagram.

1. Use Case Diagram

A use case diagram is a visual representation in the field of software engineering that illustrates the interactions between users (actors) and a system, displaying how users interact with the system.

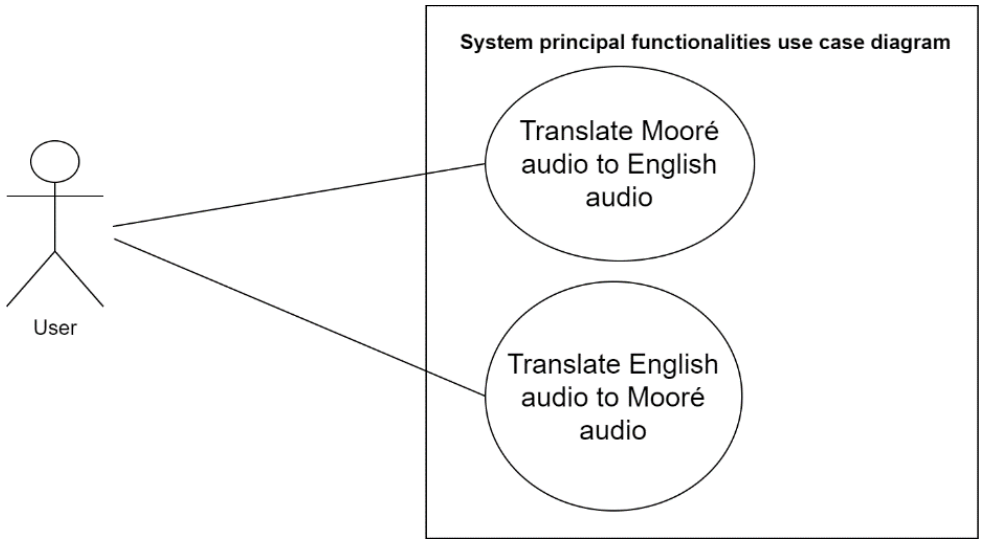

Figure 17:Use Case Diagram

2. Textual Description

It is a text that describes, in general terms, the required functionality of the use case.

Table 3:Translate from Mooré to English Textual Description

Use Case	Translate from Mooré to English
Actor	Unschooled Trader or Business Partner
Interest	Allow the user to translate from Mooré to English
Precondition(s)	The actor must have access to internet
Nominal Scenario	1. The user clicks on the flag of Burkina Faso. 2. The user records a Mooré speech and submit. 3. The system tries to recognize the submitted audio. 4. The system plays the corresponding English's audio.
Alternative Scenario	3.a The user submitted an unknow Mooré speech. The system sends an error message. The actor corrects his speech and repeats the translation process.

Table 4:Translate from English to Mooré Textual Description

Use Case	Translate from English to Mooré
Actor	Unschooled Trader or Business Partner

Interest	Allow the user to translate from English to Mooré
Precondition(s)	The actor must have access to internet
Nominal Scenario	1. The user clicks on the flag of England. 2. The user records an English speech and submit. 3. The system tries to recognize the submitted audio. 4. The system plays the corresponding Mooré's audio.
Alternative Scenario	3.a The user submitted an unknow English speech. The system sends an error message. The actor corrects his speech and repeats the translation process.

3. Sequence Diagrams

Sequence diagrams are interaction diagrams that show how operations are carried out in the system. They capture the interaction between objects in the context of a collaboration.

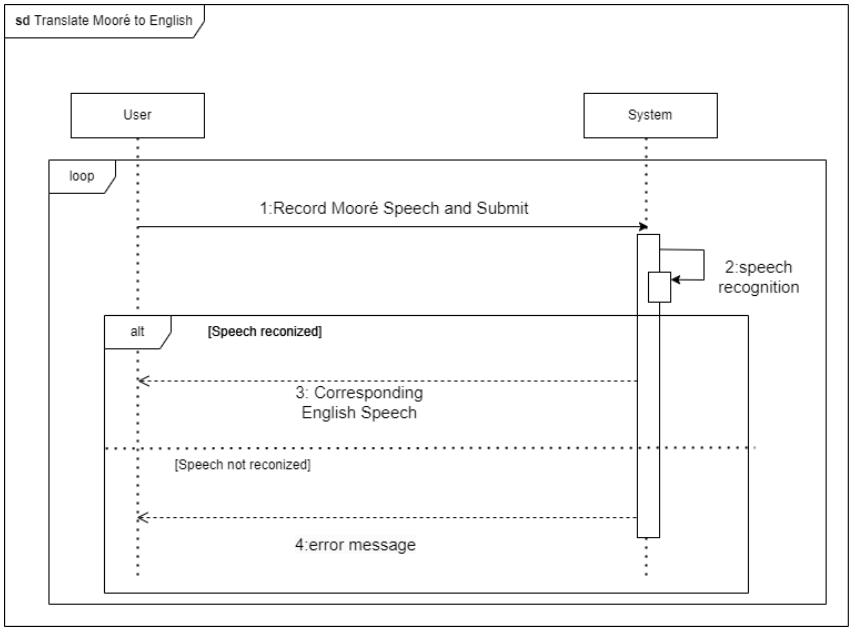

Figure 18:Sequence Diagram for Mooré to English

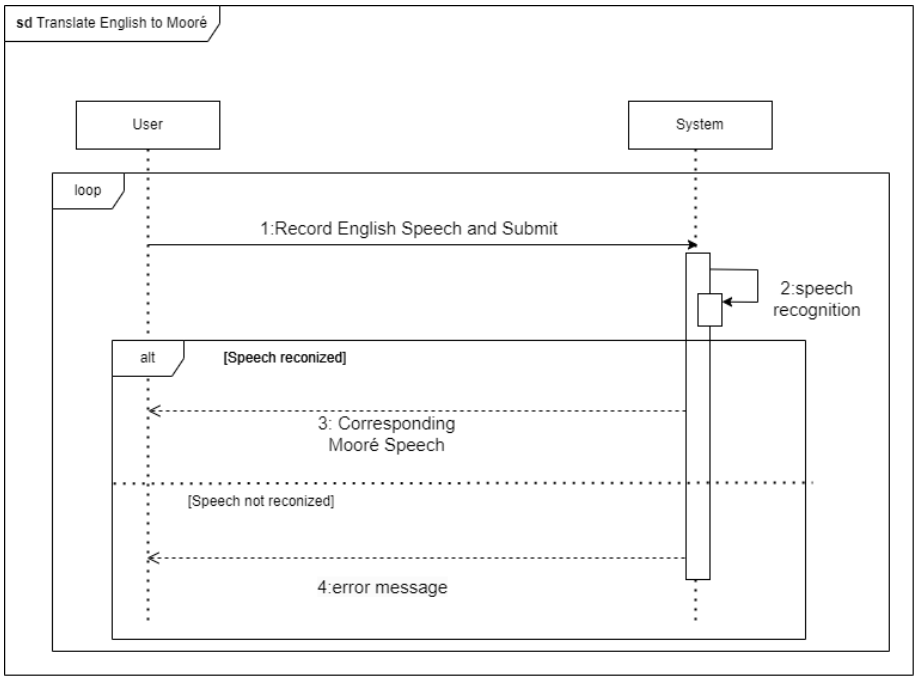

Figure 19:Sequence Diagram for English to Mooré

4. Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among them.

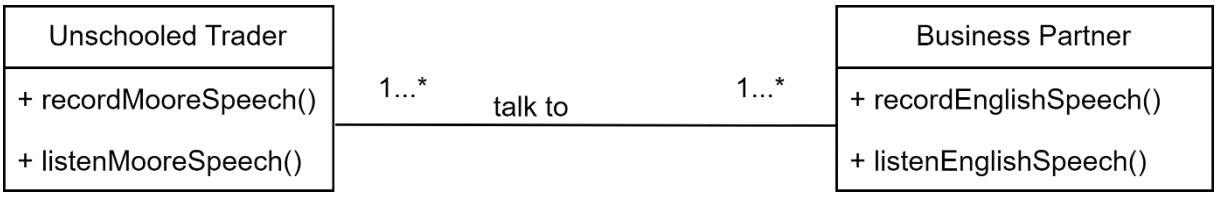

Figure 20: Class Diagram

III. Technical Study

The technical study phase concerns the technical requirements necessary to bring the translation app to life.

1. Capturing Technical Requirements

In order to satisfy the functional and technical requirements listed in the preliminary study, we used several tools and technologies and the objective of this part is to bring out all of them:

- Lenovo X1 Carbon Gen 8: Laptop with 16GB RAM, 1Terra SSD, Intel Core i7
- Windows 11: Operating System
- Visual Studio Code: for mobile and API development
- Jotform: for Data Collection
- Jupyter Notebook: for the Artificial Intelligence model training
- Android Studio: to create an Android Virtual Device
- Flutter & Dart: for the frontend programming of the application
- FastApi: for API development
- Docker: for handling environments and dependencies to avoid conflict.
- Python: to set up the API and to develop the AI model
- Tensorflow-Keras: for the AI Model building
- Draw.io: to design UML system diagrams
- Canva: to do some designs.
- Figma: to do a prototype of the application
- Voice recognizer: Python library to recognize English speech
- Google Speech to text: to convert the English voice into written text
- Librosa: it is a powerful Python library that offers a wide range of tools and functionalities for handling audio files
- MFCC (Mel Frequency Cepstral Coefficients): popular features extractor for speech signals the most use in audio recognition task.

2. Generic design

In this step, it is recommended to produce a prototype to get an idea of the interfaces and functionalities of the application. It is necessary to define mainly the colors and to produce the template of the main pages of the application.

Figure 21:Mobile App UI Design

Conclusion

At the end of this chapter, we have enough resources like the design and the necessary tools to go to our second chapter where we will implement the solution and discuss about the results.

CHAPTER 2: IMPLEMENTATION AND RESULTS

Introduction

Implementation involves putting into action the plans and strategies developed during the planning phase of a project. It is the practical execution of tasks, processes, or activities necessary to achieve the project's goals and objectives. This phase ensures that the project's design is transformed into a tangible product, service, or solution. Results are the observable, measurable, and often quantifiable changes, achievements, or deliverables that occur after the project's execution. Results can be both positive or negative and should align with the project's objectives.

I. Some codes

1. Some codes of the mobile app

The images below represent a small part of the code used to set up the main interface.

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      backgroundColor: niceColor,
      centerTitle: true,
      title: const Text(
        'No-Reesa',
        style: TextStyle(
          fontSize: 50.0,
          fontWeight: FontWeight.bold,
          color: Colors.white,
        ), // TextStyle
      ), // Text
    ), // AppBar
    backgroundColor: Colors.white,
    body: SafeArea(
      child: Column(
        children: <Widget>[
          Column(
            children: [
              const Curve(),
               Obx(() => Controller.instance.isRecording.value ||
                         Controller.instance.isRecording2.value ||
                         Controller.instance.isPlaying.value ||
                         Controller.instance.isPlaying2.value
                    ? Container()
                    : Image.asset(
                        "assets/voice.jpg",
                        width: 300,
                        height: 300,
                      )), // Image.asset // Obx
            ],
          ), // Column
```

```
Obx(() => Controller.instance.isRecording.value ||
        Controller.instance.isRecording2.value ||
        Controller.instance.isPlaying.value ||
        Controller.instance.isPlaying2.value
    ? Lottie.asset(
        "assets/Animation.json",
        repeat: true,
      )
    : Container()), // Obx
Obx( // Obx ···
Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    InkWell(
      onTap: () async {
        if (Controller.instance.isRecording.value) {
          await stopRecordingAndSave();
        } else {
          await record();
        }
      },
      child: Ink(
        decoration: BoxDecoration(
            color: niceColor,
            borderRadius: BorderRadius.circular(50)), // BoxDecoration
        child: Obx(
          () => Container(
            height: 100,
            width: 100,
            child: Controller.instance.isRecording.value
                ? const Icon(Icons.stop, color: Colors.white, size: 40,)
                :   CircleAvatar(
                    backgroundColor: niceColor,
                    radius: 68,
                    child: const CircleAvatar(
                        radius: 48, // Image radius
                        backgroundImage: AssetImage(
                            'assets/drapeau-burkina.jpg'), // AssetImage
                      ), // CircleAvatar
                    ), // CircleAvatar
          ), // Container
        ), // Obx
      ), // Ink
    ), // InkWell
    InkWell(
      onTap: () async {
        if (Controller.instance.isRecording2.value) {
          await stopRecordingAndSave2();
        } else {
          await record2();
        }
      },
      child: Ink(
        decoration: BoxDecoration(
            color: niceColor,
            borderRadius: BorderRadius.circular(50)), // BoxDecoration
        child: Obx(() => Container(
          height: 100,
          width: 100,
          child:
                Controller.instance.isRecording2.value
                    ? const Icon(Icons.stop, color: Colors.white, size: 40,)
                    :    CircleAvatar(
                    backgroundColor: niceColor,
                    radius: 68,
```

```
                        color: niceColor,
                        borderRadius: BorderRadius.circular(50)), // BoxDecoration
                    child: Obx(() => Container(
                      height: 100,
                      width: 100,
                      child:
                            Controller.instance.isRecording2.value
                                ? const Icon(Icons.stop, color: Colors.white, size: 40,)
                                :    CircleAvatar(
                              backgroundColor: niceColor,
                              radius: 68,
                              child: const CircleAvatar(
                                  radius: 48, // Image radius
                                  backgroundImage: AssetImage(
                                      'assets/drapeau-anglais.jpg'), // AssetImage
                              ), // CircleAvatar
                            ), // CircleAvatar
                          )), // Container // Obx
                    ), // Ink
                  ), // InkWell
                ],
              ), // Row
            ], // <Widget>[]
          ), // Column
        ), // SafeArea
      ); // Scaffold
    }
  }
```

2. Some codes of the AI Model development

The code bellow shows the different libraries that we import for training our model.

```
[1]: import os
     import librosa
     import numpy as np
     import pandas as pd
     import tensorflow as tf
     import matplotlib.pyplot as plt
     import librosa.display
     from tensorflow.keras.layers import Input, Flatten, Dense, Dropout,Conv1D,Conv2D, MaxPooling1D, Bidirectional, LSTM
     from tensorflow.keras.models import Model, load_model
     from tensorflow.keras.optimizers import Adam
     from sklearn.model_selection import train_test_split
     from sklearn.preprocessing import LabelEncoder
     from tensorflow.keras.utils import to_categorical
     from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
     from tensorflow.image import resize
     import sounddevice as sd
     import wavio
```

This function was used to extract voice features to train our model.

```
[2]: def extract_features(file_path, n_mfcc=42, max_pad_len=100):
         audio, sample_rate = librosa.load(file_path, res_type='kaiser_fast')
         mfccs = librosa.feature.mfcc(y=audio, sr=sample_rate, n_mfcc=n_mfcc)
         pad_width = max_pad_len - mfccs.shape[1]
         if pad_width > 0:
             mfccs = np.pad(mfccs, pad_width=((0, 0), (0, pad_width)), mode='constant')
         else:
             mfccs = mfccs[:, :max_pad_len]
         return mfccs.T  #
```

The code here was use to split our dataset into training and validation sets.

```
# Convert to numpy arrays
X = np.array(extracted_features)
y = np.array(labels)

# split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# encode the labels
y_train = to_categorical(y_train, num_classes=len(classes))
y_test = to_categorical(y_test, num_classes=len(classes))
```

3. Some codes of the API

This is the code used to recognize English speech and answer with the corresponding Mooré voice.

```
@app.post("/predict/english_to_moore")
async def predict_english_to_moore(file: UploadFile = File(...)):
    try:
        audio_bytes = await file.read()
        # Recognize speech using SpeechRecognition
        recognizer = sr.Recognizer()
        audio_data = BytesIO(audio_bytes)
        with sr.AudioFile(audio_data) as source:
            audio_content = recognizer.record(source)
            recognized_text = recognizer.recognize_google(audio_content).lower()
            moore_text = dic_english_moore.get(recognized_text, "Translation not found")
            moore_audio_path = f"moore-audios/{moore_text}.wav"
            if not os.path.exists(moore_audio_path):
                raise HTTPException(status_code=404, detail="Audio file not found for the translation")
        return FileResponse(moore_audio_path, media_type='audio/wav', filename='audio.wav')
    except sr.UnknownValueError:
        raise HTTPException(status_code=400, detail="Could not understand the audio")
    except sr.RequestError as e:
        raise HTTPException(status_code=500, detail=f"Could not request results from the speech recognition service; {e}")
    except Exception as e:
        raise HTTPException(status_code=500, detail=f"Error during prediction: {e}")
```

II. Results

1. Results of the AI model

a. Results of the model trained using the initial dataset

In order to select the best algorithm for our project, we tested five algorithms that are: the CNN, the LSTM, the BILSTM, the CNN-LSTM and the CNN-BILSTM.

Here are the results that we got after **100 epochs**:

Table 5:Results with different algorithms used to train the model with the initial dataset

Algorithm	CNN	LSTM	BILSTM	CNN-LSTM	CNN-BILSTM
Accuracy (%)	21.76	76.73	73.73	75.98	79.92

This table shows that the CNN-BILSTM give the best accuracy, then we will just continue our study with this algorithm.

- **Loss Curve of the model trained using the initial dataset**

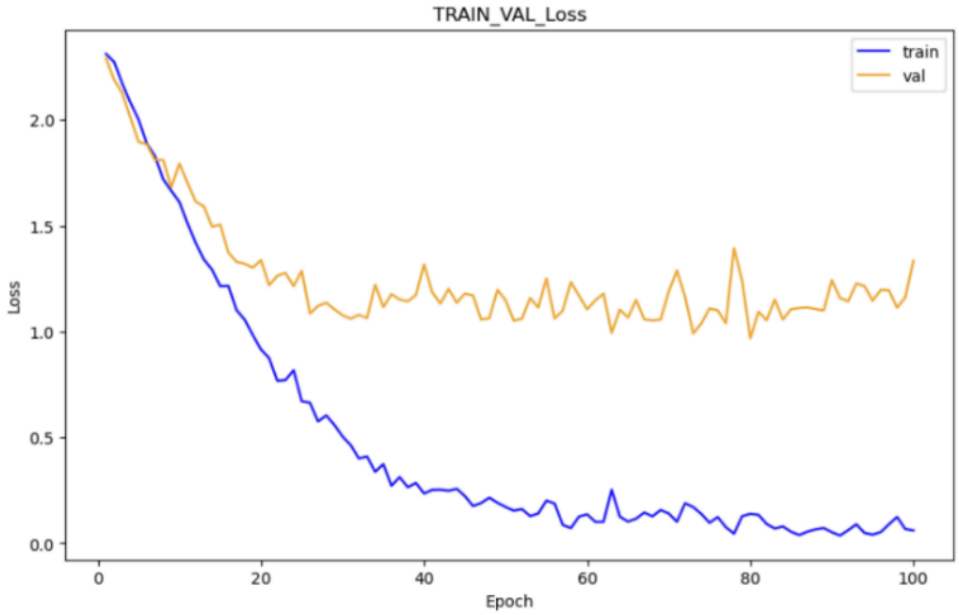

Figure 22:Loss Curve of the model trained with initial dataset

- **Accuracy Curve of the model trained using the initial dataset**

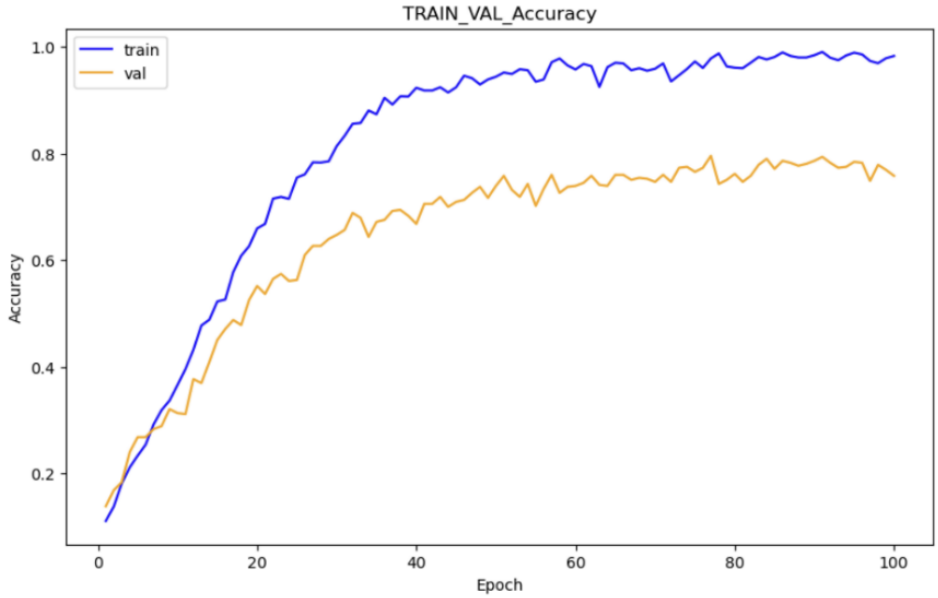

Figure 23:Accuracy Curve of the model trained with initial dataset

- **Confusion Matrix of the model trained using the initial dataset**

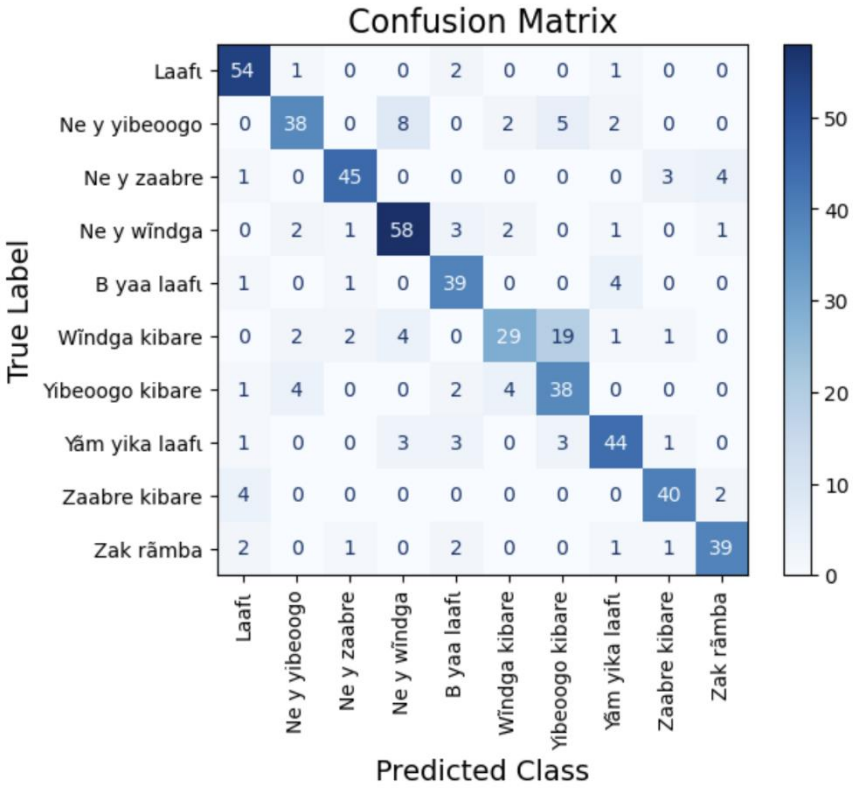

Figure 24:Confusion matrix of the model trained with initial dataset

- **Scores per classes using F1-score of the model trained using the initial dataset**

```
F1 Score for class Laafɩ: 0.89
F1 Score for class Ne y yibeoogo: 0.75
F1 Score for class Ne y zaabre: 0.87
F1 Score for class Ne y wĩndga: 0.82
F1 Score for class B yaa laafɩ: 0.81
F1 Score for class Wĩndga kibare: 0.61
F1 Score for class Yibeoogo kibare: 0.67
F1 Score for class Yãm yika laafɩ: 0.81
F1 Score for class Zaabre kibare: 0.87
F1 Score for class Zak rãmba: 0.85
```

Figure 25: Scores per classes of the model trained with initial dataset

b. Results of the model trained using the augmented dataset

- **Loss Curve of the model trained using the augmented dataset**

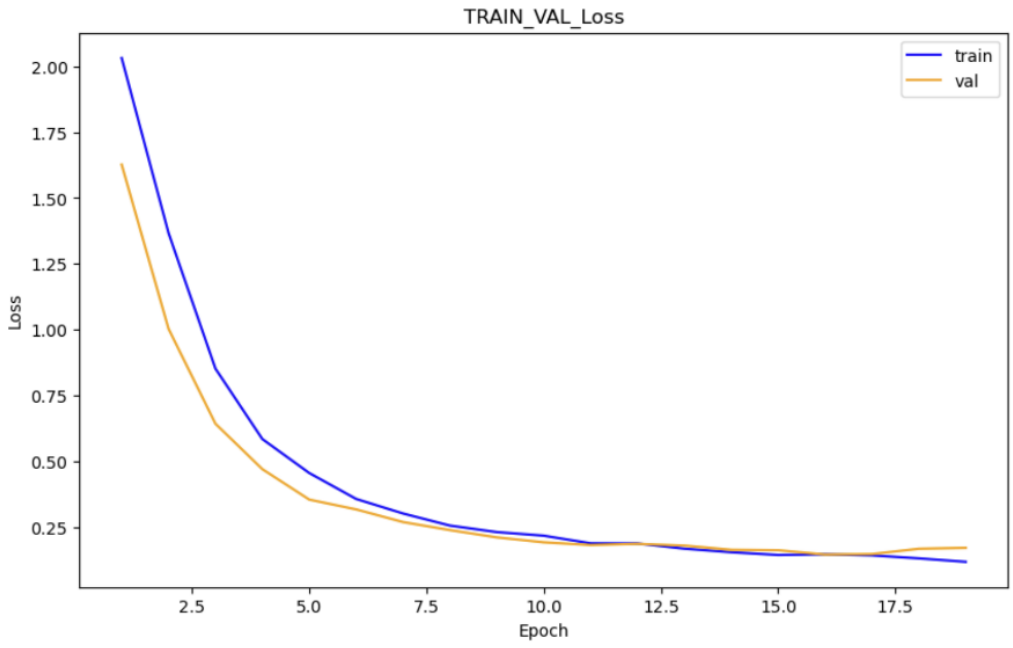

Figure 26: Loss Curve of the model trained with augmented dataset

- **Accuracy curve of the model trained using the augmented dataset**

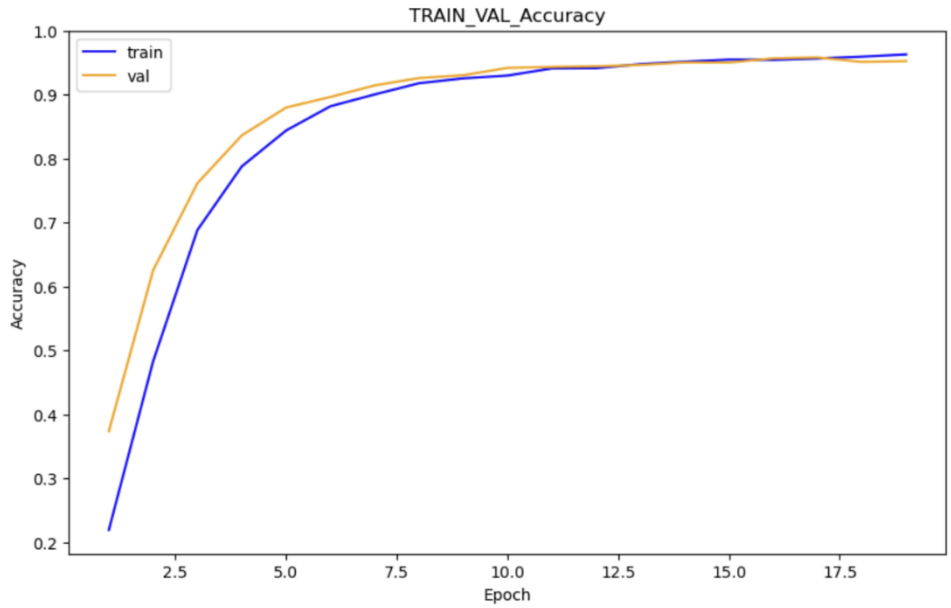

Figure 27: Accuracy Curve of the model trained with augmented dataset

- **Confusion Matrix of the model trained using the augmented dataset**

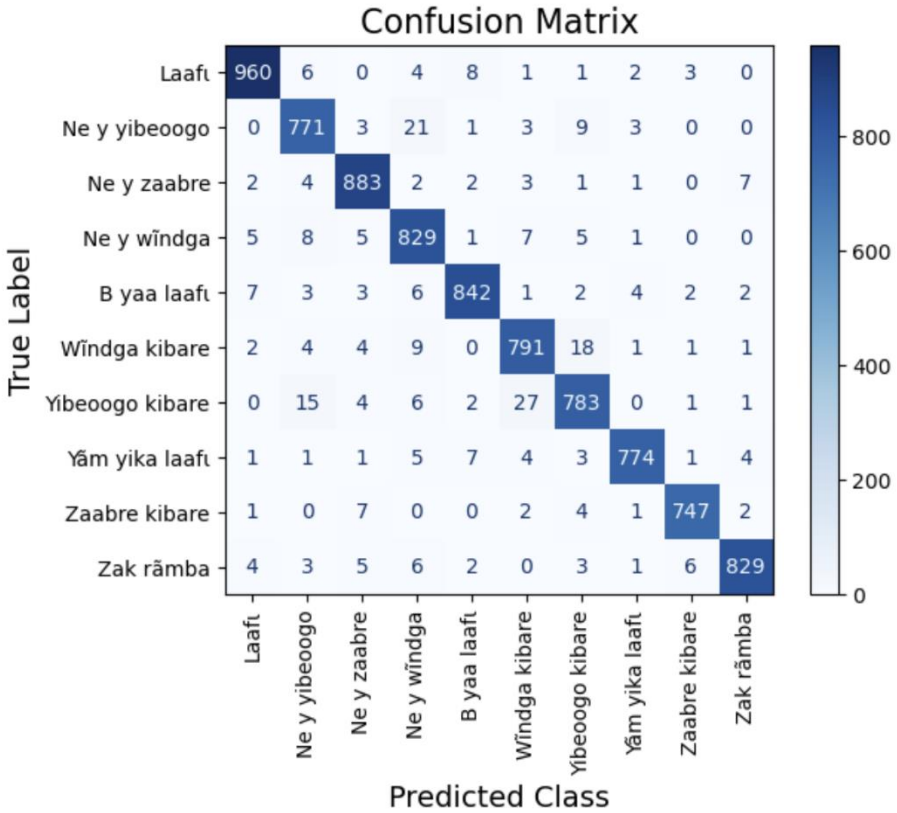

Figure 28:Confusion Matrix of the model trained with augmented dataset

- **Scores per classes using F1-score of the model trained using the augmented dataset**

```
F1 Score for class Laafɩ: 0.98
F1 Score for class Ne y yibeoogo: 0.95
F1 Score for class Ne y zaabre: 0.97
F1 Score for class Ne y wĩndga: 0.95
F1 Score for class B yaa laafɩ: 0.97
F1 Score for class Wĩndga kibare: 0.95
F1 Score for class Yibeoogo kibare: 0.94
F1 Score for class Yãm yika laafɩ: 0.97
F1 Score for class Zaabre kibare: 0.98
F1 Score for class Zak rãmba: 0.97
```

Figure 29:Scores per classes of the model trained with augmented dataset

2. Results of the Mobile App Development

These are some pictures showing the interfaces of our mobile app.

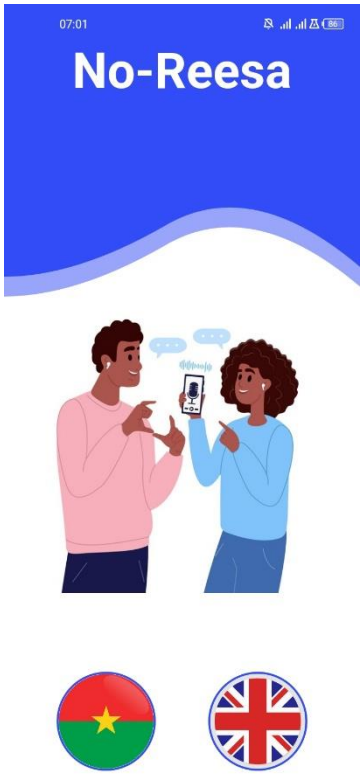

Figure 30:Home Page of the mobile App

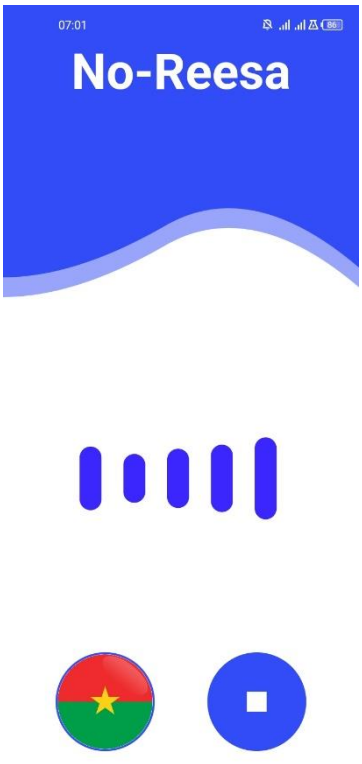

Figure 31:UI when the App is listening to an English Speech

Figure 33::UI when the App is listening to Mooré Speech

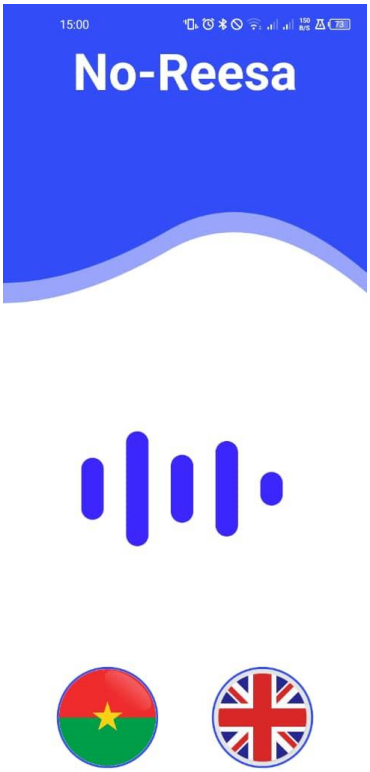

Figure 32:UI when the app is playing an audio

III. Discussions

1. Discussions on the results of the model trained with initial dataset

Based on loss curve, we can remark that the training loss is decreasing, the validation loss plot suggests that the model is overfitting. In addition, while the model shows strong learning capability through the accuracy curve on the training set, its performance on the validation set indicates also an overfitting.

Through the confusion matrix, we can observe that the model seems to struggle differentiating "Wĩndga kibare" and " Yibeoogo kibare" as shown by the 19 misclassifications.

We can conclude that the results of this model are not satisfactory it need improvement. Let us now see the effect of using augmented dataset to train the model.

2. Discussions on the results of the model trained with augmented dataset

The loss and accuracy curves suggest that the model training is progressing well. The rapid initial decrease of loss curve and increase of accuracy curve followed by a stabilisation indicates effective learning. Since the validation loss follows the training loss closely and the validation accuracy follows the training accuracy closely, all eventually stabilize, the model generalizes well to the validation set without overfitting.

The confusion matrix shows that, the classifier performs well across most classes, with high true positive rates and relatively low misclassifications.

Some classes show particularly strong performance while the others have slightly higher misclassifications, indicating areas where the classifier may need improvement.

After analysing these results, we can say that the augmentation of the dataset helped to improve the model accuracy but more data will help to level up the model for better results. In addition, the strong performance of some classes may be caused by the unbalanced size of the data per classes.

IV. Financial evaluation

1. Presentation of the COCOMO Model

The COCOMO method (COnstructive COst MOdel) is a model developed by Barry W. Boehm in the 1980s allowing to define an estimation of the effort to be provided in a software development and the duration that this one will take according to the allocated resources.

2. COCOMO model formula

It is necessary to have the assumed number of program lines in source language, apart from possible comments. The model allows to obtain the charge of the realization in person-months, as well as the normal delay recommended to reduce additional risks. This gives an indication of the average size of the team that is equal to load/time.

The formulas for calculating load and delay are as follows:

- Charge in person-months= a $(KLSL)^b$;
- Normal lead time in months = c (charge in person-months) d with KLSL = Kilo of Source Line Delivered.

The parameters 'a', 'b', 'c' and 'd' take different values depending on the project category. Indeed, the statistical analysis led to distinguish three types of projects: simple, medium and complex. A project is simple if the software has less than 50,000 instructions, if the specifications are stable and the development is done by a small team.A project is medium if the software has between 50,000 and 300,000 instructions.A project is complex if the software has more than 300,000 instructions and if a large team is expected; it often applies to a new domain. The values of the parameters are as follows:

Table 6:COCOMO model charge and time values based on the type of project

Type of Project	Charge	Time
Simple	2,4 (KLSL) ^{1,05}	2,5(Charge) ^{0,38}
Medium	3 (KLSL) ^{1,12}	2,5(Charge) ^{0,35}
Complex	3,6 (KLSL) ^{1,2}	2,5(Charge) ^{0,32}

3. Development cost

In the context of the application's realization and referring to the previous table, this is a medium project with an estimated number of instructions of 80,457 lines of code and an estimation of 1500 FCFA per hour for 12 hours per day over 3 months. A git script was used to estimate the number of lines in the project.

Application:

Load: 3 *(80.457) $^{1.12}$ =408.65 person-month

Normal lead time: 2.5*(408.65) $^{0.35}$ =20.5 month

Average team size: 408.65 / 20.5 = 19.93 persons

Normal total Cost: (408.65 *1500*30*12) *3=662.013.000 FCFA.

Conclusion

In this chapter we gave an overview on how we implemented the project, discussed about the results that we got and finished by doing a financial evaluation of the solution. Let’s now do a general conclusion to conclude our whole work.

General Conclusion and Perspectives

In this document, we worked on the development of a voice translation mobile app using artificial intelligence: case of Mooré and English in order to contribute to unleash language barrier between unschooled traders and their business partners. This was a challenging project due to the fact that the Mooré language is a low resource language. To do so, firstly, we explained deeply the origin of this topic then we tried to organize our work using the 2TUP process. Secondly, we exanimated the requirements of this kind of project and select the right tools to implement it. Finally, we did an implementation of the project, gave the results that we got and then discussed them before ending with financial evaluation.

We are proud to see that despite the challenges that we met during the development of this project, we were able to get good results that show that it is possible to do a voice translation from our local languages to the popular ones like English. For sure this is not sufficient but can be the basic for an upcoming voice translation app applied to our local languages. This is not the end of our work, in the future we planned to:

- Add more classes to our datasets to increase the scope of translation.
- Add other local languages like Dioula, Fulfuldé...
- Make the difference between men, women and children voices.
- Do a speech to text instead of predicting the classes.
- Implement a text-to-text model from our local languages to English.
- Implement a text to speech for our local languages.
- Merge all these models together to do a real automatic speech to speech system.
- Integrate the model in earphones for automatic translation.

Bibliography

[1] "plaquette_resultats_definitifs_rgph_2019_-_revu_30-12-2022.pdf." Accessed: Aug. 04, 2024. [Online]. Available: http://cns.bf/IMG/pdf/plaquette_resultats_definitifs_rgph_2019_-_revu_30-12-2022.pdf

[2] "Burkina Faso | World Bank Gender Data Portal." Accessed: May 21, 2024. [Online]. Available: https://genderdata.worldbank.org/en/economies/burkina-faso

[3] C. A. Montgomery *et al.*, "MACHINE-AIDED VOICE TRANSLATION (MAVT)".

[4] I. Lezhenin, N. Bogach, and E. Pyshkin, "Urban Sound Classification using Long Short-Term Memory Neural Network," presented at the 2019 Federated Conference on Computer Science and Information Systems, Sep. 2019, pp. 57–60. doi: 10.15439/2019F185.

[5] C. C. Emezue and F. P. B. Dossou, "FFR v1.1: Fon-French Neural Machine Translation," presented at the Proceedings of the Fourth Widening Natural Language Processing Workshop, Jul. 2020, pp. 83–87. doi: 10.18653/v1/2020.winlp-1.21.

[6] L. Martinus, J. Webster, J. Moonsamy, M. S. Jnr, R. Moosa, and R. Fairon, "Neural Machine Translation for South Africa's Official Languages," May 08, 2020, *arXiv*: arXiv:2005.06609. Accessed: Aug. 12, 2024. [Online]. Available: http://arxiv.org/abs/2005.06609

[7] G. Hacheme, "English2Gbe: A multilingual machine translation model for {Fon/Ewe}Gbe," Dec. 13, 2021, *arXiv*: arXiv:2112.11482. Accessed: Aug. 12, 2024. [Online]. Available: http://arxiv.org/abs/2112.11482

[8] "CONGO FAISSAL MEMOIRE DEFINITIF.pdf."

[9] "Rapport_master_final_ouily.pdf."

[10] "memoireMaimouna2.pdf."

[11] "Rapport_projet_tuteure.pdf."

[12] "Speech_Noise_detection.pdf."

[13] "What is Artificial Intelligence (AI)? | IBM." Accessed: Jun. 04, 2024. [Online]. Available: https://www.ibm.com/topics/artificial-intelligence

[14] "What Is NLP (Natural Language Processing)? | IBM." Accessed: Jun. 06, 2024. [Online]. Available: https://www.ibm.com/topics/natural-language-processing

[15] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental Sound Recognition With Time–Frequency Audio Features," *IEEE Trans. Audio Speech Lang. Process.*, vol. 17, no. 6, pp. 1142–1158, Aug. 2009, doi: 10.1109/TASL.2009.2017438.

[16] S. Chachada and C.-C. J. Kuo, "Environmental sound recognition: A survey," in *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Oct. 2013, pp. 1–9. doi: 10.1109/APSIPA.2013.6694338.

[17] "Detection and classification of acoustic scenes and events: An IEEE AASP challenge | IEEE Conference Publication | IEEE Xplore." Accessed: Jul. 10, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/6701819

[18] "Kons and Toledo-Ronen - 2013 - Audio event classification using deep neural netwo.pdf."

[19] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2015, pp. 1–6. doi: 10.1109/MLSP.2015.7324337.

[20] J. Salamon and J. P. Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, Mar. 2017, doi: 10.1109/LSP.2017.2657381.

[21] V. Boddapati, A. Petef, J. Rasmusson, and L. Lundberg, "Classifying environmental sounds using image recognition networks," *Procedia Comput. Sci.*, vol. 112, pp. 2048–2056, Jan. 2017, doi: 10.1016/j.procs.2017.08.250.

[22] Y. Wang, L. Neves, and F. Metze, "Audio-based multimedia event detection using deep recurrent neural networks," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2016, pp. 2742–2746. doi: 10.1109/ICASSP.2016.7472176.

[23] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and Classification of Acoustic Scenes and Events," *IEEE Trans. Multimed.*, vol. 17, no. 10, pp. 1733–1746, Oct. 2015, doi: 10.1109/TMM.2015.2428998.

[24] J. Sang, S. Park, and J. Lee, "Convolutional Recurrent Neural Networks for Urban Sound Classification Using Raw Waveforms," in *2018 26th European Signal Processing Conference (EUSIPCO)*, Rome: IEEE, Sep. 2018, pp. 2444–2448. doi: 10.23919/EUSIPCO.2018.8553247.

[25] "Collecte de vocaux en langue mooré." Accessed: Jul. 13, 2024. [Online]. Available: https://form.jotform.com/233633766919065