DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND RESEARCH
UNIVERSITY of SAAD DAHLEB BLIDA
Faculty of Sciences
Computer Science department

MASTER'S THESIS
In computer science

Option :Natural Language Processing
And : Software engineering

Topic :

Proposition of a neural solution to translate sign language into Algerian dialect

Produced by
Chenouia Soumia
Benmoussa Amira

Supervised by
Melyara Mezzi

Acknowledgments

We would like to begin by expressing our heartfelt gratitude to Allah for His blessings, guidance, and the strength He has bestowed upon us throughout this project. His mercy and divine support have been instrumental in our success, and we are deeply grateful.

first of all, we extend our sincere appreciation to our supervisor, **Mrs. Melyara Mezzi**, for her invaluable guidance, expertise, and continuous support during the course of this thesis. Her dedication to our academic development and her unwavering belief in our abilities have been truly inspiring.

We would like to thank the members of the **jury** for kindly participating in the evaluation of this work.

We would also like to acknowledge the contribution of **Mekid Hamza,** whose assistance has greatly enhanced the quality of our work.

Furthermore, we would like to express our deepest gratitude to our **parents and families** for their unwavering support, encouragement, and sacrifices. Their love, understanding, and constant motivation have been the driving force behind our achievements, and we are forever indebted to them.

We would like to extend our thanks to all our **friends** and colleagues who have supported us throughout this journey. Their encouragement, discussions, and feedback have been immensely valuable, and we are grateful for their presence in our lives.

Dedications

I want to take a moment to express my deepest appreciation to **myself**. I celebrate this moment and the incredible individual I have become. then I want to dedicate this Work to:

My mother , who has shown me the true meaning of love, a love that is boundless, unconditional, and infinite. Her love has been the constant force that gives me the courage to face life's obstacles, the warmth that envelops me during my darkest hours, and the light that illuminates my path.

Zineb who has been more than just a sister to me. She is my confidante, my best friend, and a constant source of inspiration. Zineb is the most amazing sister anyone could ask for, and I am incredibly lucky to have her in my life.

Oulfa, my soulmate, who has been there for me throughout my academic journey, pushing me to reach my full potential and never giving up on me. Your unwavering belief in my abilities and your relentless support have been instrumental in my success.

Mohamed, as a token of my appreciation for your friendship , you have always believed in me, even when I doubted myself. you have had a positive impact on my life. Your presence has made a significant difference, and I am truly fortunate to have you as my best friend.

Sidali, who has always been there for me, offering a listening ear, words of wisdom, and a shoulder to lean on. his unwavering belief in my abilities and his constant support have given me the confidence to face challenges and pursue my dreams.

ITC members for their unwavering support and invaluable assistance Thank you for being an integral part of my academic journey.

Soumia

Dedications

First and first I thanks god for giving me the strength to go throw these previous 5 years with all its ups and downs.and giving me opportunities to explore many sides of me .

Second, I didicate this work to myself,the only one who diden't gives up on me .

I didicate this work to my parents :my mother who always challenge me even if sometimes its hurt and my father the one that no word can describe him and the one who tougth me how to dream.

A didicate full of flowers to my sister who keep adding suger to my life since the day she came to life.

I didicate this work to my lovely friends lydia azzouz,sara benmokadem and monssif brahim the ones who made this experience unforgatebal.

I didicate this work to my lovely talented loubna menia,saber menia and wafa who theeir presence in my life means a lot.

I didicate this work to my ex-binome manel amrouch who gaved me the support needed in my darkest days .

I didicate this work to my dear Sadji Zigadi ,Amir Oustelmane and nowi for their trust .

I didicate this work to club bibliophiles where everything started,IT Community club ,GDG les rosier,WTM les rosier ,AIESEC in Algeria and AIESEC in Finland for all what I learned and still learning from them.

I didicate this work to all the students and professors of the institute of aeronautics .

I didicate this work to all the teachers and professors who still belive in youth potential.

An unfinit didicate to everyone who still dreaming in this coutry and still fighting for them.

I didicate this work to you the reader of this thesis ....

Amira

Abstract

Sign language is a distinct form of communication essential for various segments of society. It encompasses a diverse range of signs, each characterized by variations in hand shape, motion profile, and the positioning of hands, face, and body parts. Consequently, visual sign language recognition represents a complex area of research within computer vision. In recent years, significant advancements have been made, mainly through using deep learning approaches, as proposed by various researchers.

This work focuses explicitly on translating American Sign Language (ASL) into the Algerian dialect, with the overarching goal of bridging the communication gap between the ASL-based deaf community and speakers of the Algerian dialect. The project consists of two primary components. Firstly, a sign language recognition phase, where two models have been developed to detect ASL signs in static images and in real-time accurately. Secondly, a translation phase that employs a word to word translation techniques to convert the recognized signs into the Algerian dialect.

key words:

Sign Language, Algerian Dialect, Machine Translation ,Computer vision,Deep learning.

الملخص

لغة الإشارة هي شكل متميز من أشكال التواصل التي لها أهمية كبيرة لمختلف شرائح المجتمع. وهي تشمل مجموعة متنوعة من العلامات ،كل منها يتميز باختلافات في شكل اليد ، ومظهر الحركة ، ووضع اليدين والوجه وأجزاء الجسم. وبالتالي ، فإن التعرف على لغة الإشارة المرئية يمثل مجالًا معقدًا من البحث ضمن رؤية الكمبيوتر. في السنوات الأخيرة ، تم إحراز تقدم كبير ، لا سيما من خلال استخدام مناهج التعلم العميق ، على النحو الذي اقترحه العديد من الباحثين.

يركز هذا العمل بشكل خاص على ترجمة لغة الإشارة الأمريكية إلى اللهجة الجزائرية ، مع الهدف الشامل المتمثل في سد فجوة التواصل بين مجتمع الصم القائم على لغة الإشارة الأمريكية والمتحدثين باللهجة الجزائرية. يتكون المشروع من عنصرين رئيسيين. أولاً ، مرحلة التعرف على لغة الإشارة ، حيث تم تطوير نموذج لاكتشاف إشارات بدقة في الصور الثابتة وفي الوقت الفعلي. ثانيًا ، مرحلة الترجمة التي تستخدم تقنيات الترجمة الآلية لتحويل الإشارات المعترف بها إلى اللهجة الجزائرية.

الكلمات المفتاحية :
لغة الإشارة ، اللهجة الجزائرية ، الترجمة الآلية ، الرؤية الحاسوبية ، التعلم العميق.

Résumé

La langue des signes est une forme distincte de communication essentielle pour divers segments de la société. Il englobe une gamme variée de signes, chacun caractérisé par des variations dans la forme de la main, le profil de mouvement et le positionnement des mains, du visage et des parties du corps. Par conséquent, la reconnaissance visuelle du langage des signes représente un domaine de recherche complexe au sein de la vision par ordinateur. Ces dernières années, des progrès significatifs ont été réalisés, principalement grâce à l'utilisation d'approches d'apprentissage en profondeur, telles que proposées par divers chercheurs.

Ce travail se concentre explicitement sur la traduction de la langue des signes américaine (ASL) dans le dialecte algérien, avec l'objectif primordial de combler le fossé de communication entre la communauté sourde basée sur l'ASL et les locuteurs du dialecte algérien. Le projet comprend deux volets principaux. Premièrement, une phase de reconnaissance de la langue des signes, où deux modèles ont été développés pour détecter les signes ASL dans des images statiques et en temps réel avec précision. Deuxièmement, une phase de traduction qui utilise des techniques de traduction automatique pour convertir les signes reconnus dans le dialecte algérien.

Mots clés:
Langue des signes, dialecte algérien, traduction automatique, vision par ordinateur, apprentissage en profondeur.

Contents

List of Figures

List of Tables

List of Abbreviations

ALG algerian

API Application Programming Interface

ASL American Sign Language

BERT Bidirectional Encoder Representations from Transformers

CNN convolutional neural network

DNN Deep Neural Networks

EOS end-of-sentence symbol

GPU Graphics Processing Unit

GPT Generative Pre Training Transformer

LSTM Long short term memory

LSA Algerian Sign Language

LSF French Sign Language

ML machine learning

MSA Modern Standard Arabic

MT Machine translation

NLP Natural language processing

NLG Natural Language Generation

NLU Natural Language Understanding

NN neural network

NMT Neural Machine Translation

PANet Path Aggregation Network

RNN Recurrent Neural Ntworks

RGB Red Green Blue

UML Unified Modeling Language

WASL World level American Sign Language

WFD The World Federation of the Deaf

YOLO You Only Look Once

General Introduction

Work context

Effective communication plays a crucial role in our daily lives and holds particular significance for individuals with hearing impairments. Sign language is a visual communication used by deaf individuals to express themselves and interact with others. However, the understanding and interpretation of sign language can vary across regions, which can pose challenges in translation, mainly when translating signs into a specific language or dialectal variant.

This research focuses on translating Sign Language into the Algerian dialect. It aims to develop a system capable of recognizing and translating signs using a specific approach tailored to the Algerian dialect.

Research Problem

The main challenge of this research lies in accurately and fluently translating sign Language into the Algerian dialect. Sign languages differ from one country to another and from one region to another, and the cultural, linguistic, and gestural differences between spoken languages and sign languages can lead to interpretation errors and loss of meaning during translation. Therefore, it is essential to choose one sign language to focus on and develop a precise sign recognition model and a translation system that suit the specificities of the Algerian dialect, enabling effective communication between deaf individuals and speakers of the Algerian dialect.

Objectives

The objectives of this research are as follows:

- Develop a reliable and accurate model for recognizing American Sign Language. This entails employing advanced machine learning and computer vision techniques to capture and interpret the distinctive gestures and movements of sign language.
- Design a translation system capable of converting recognized signs into text in the Algerian dialect. This requires a comprehensive understanding of the linguistic and cultural nuances of the Algerian dialect, along with the establishment of appropriate translation rules.
- Evaluate and enhance the system's performance by measuring the accuracy of sign recognition and the quality of translation. This evaluation will be conducted in collaboration with native users of both American Sign Language and the Algerian dialect to ensure the relevance and effectiveness of the developed system.

Organization of the Thesis

This thesis is divided into several sections that address different phases of the research . The first chapter presents a literature review of prior work on sign language translation, Algerian dialect, gesture recognition, and machine translation systems. It highlights existing gaps and research opportunities in the field. Then, the second chapter describes the methodology employed to develop the American Sign Language recognition model and the translation system for the Algerian dialect. It outlines the data collection and processing steps, the machine learning techniques utilized, and the linguistic approaches employed for translation.

In the third chapter, we present the results obtained from experimentation and system evaluation, analyze the sign recognition model's performance and assess the translation quality achieved.

Finally, the general conclusion discusses the research contributions, identified limitations, and future improvement avenues. It concludes by emphasizing the significance of this work in the field of communication for deaf individuals and the prospects it opens for further research and development.

Chapter 1

Sign language Translation

1.1 Introduction

Deafness and hearing loss individuals are part of our community, as there are more than 70 million deaf people worldwide [17]. Unless their disability, their human rights should be respected [18]:
Article 3: Everyone has the right to life, liberty, and the security of a person.
Article 6: Everyone has the right to recognition everywhere as a person before the law.
Article 9:Everyone has the right to freedom of opinion and expression.

To protect their rights, organizations have been created, such as The World Federation of the Deaf [19], and their sign languages have been developed through time. Today, we have 300 sign languages worldwide. Many facilities have been created [17] as shown in Figure 1.1 , but not all hearing persons understand sign language, so life is still hard for unhearing individuals.

Projects like Intel Omnibridge [1], and Hear-o App [2], have been created to solve this issue. Our community [20] is concerned with 71800 unhearing individuals. We took this challenge, and throw this chapter, we will discuss it in depth.

1

Intel Omnibridge https://omnibridge.ai/

2

Hear-o App https://www.indiegogo.com/projects

1.2 deafness and hearing loss

We will begin with a brief definition and then delve deeper :

1.2.1 Definition

"A person who is not able to hear as well as someone with normal hearing – hearing thresholds of 20 dB or better in both ears – is said to have hearing loss. Hearing loss may be mild, moderate, severe, or profound. It can affect one ear or both ears and leads to difficulty in hearing conversational speech or loud sounds." [21]

Approximately 3 in 100 babies are born with permanent hearing loss, making hearing loss one of the most common birth defects in America,and nearly 80% of people with disabling hearing loss live in low- and middle-income countries.[21] Deaf' people mostly have profound hearing loss, which implies very little or no hearing. They often use sign language for communication.

1.2.2 Some Causes

Here we present some leading causes of deafness and hearing loss [21]:

- Genetic factors including hereditary and non-hereditary hearing loss.
- Birth asphyxia [3],
- Chronic diseases.
- Smoking.
- Loud noise/loud sounds.

1.2.3 World hearing day

World Hearing Day is held on 3 March each year to raise awareness on how to prevent deafness and hearing loss and promote ear and hearing care across the world [22] . Figure 1.2 shows a poster from this year's word hearing day.

[3]

Birth asphyxia a lack of oxygen at the time of birth [21].

Figure 1.1: A video call using sign language.

Figure 1.2: World Hearing Day poster for 2023. [1].

1.3 Sign language

Sign language is a visual-spatial language that relies on a combination of hand gestures, facial expressions, and body movements to convey meaning and communicate. This section aims to provide a comprehensive definition and explanation of this unique form of language.

1.3.1 Definition

Sign language «is manual communication commonly used by people who are deaf. Sign language is not universal; people who are deaf from different countries speak different sign languages. The gestures or symbols in sign language are organized linguistically. Each gesture is called a sign. Each sign has three distinct parts: the handshape, the position of the hands, and the movement of the hands» [23].

Figure 1.3 shows Deaf and hard-of-hearing students participating in a lesson at a school for deaf people in Iraq.

Figure 1.3: Deaf and hard-of-hearing students participating in a lesson at a school for deaf people in Iraq[2].

1.3.2 Sign language history

People with hearing impairments have been marginalized for millennia because it was once thought that learning a language required hearing it spoken.[24] For example, under Roman law, people who were born deaf were not allowed to sign a will because it was assumed that they knew nothing and could not have learned to read or write.

Aristotle was the first to have a claim recorded about the deaf. His theory was that people can only learn through hearing spoken language. Deaf people were therefore seen as being unable to learn or be educated at all [25].

In the Renaissance, resistance to this bias first emerged. Pedro Ponce de León, a Benedictine monk from Spain who lived in the 16th century, is the first person to be given credit for developing formal sign language for the deaf. He wasn't the first to utilize sign language, though. To communicate with neighboring tribes and to enable trade with Europeans, Native Americans employed hand signals. They were used by Benedictine monks to communicate during their regular hours of quiet [24].

Today, there are more than 300 different sign languages in the world, spoken by more than 72 million deaf or hard-of-hearing people worldwide [26].

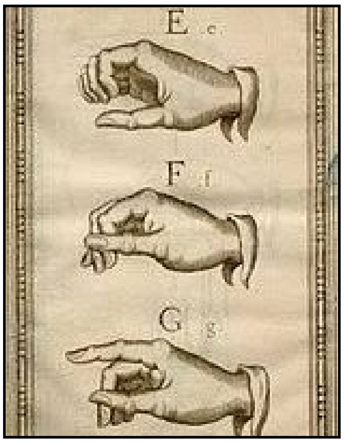


Figure 1.4: Juan Pablo Bonet's 1620 Reduction of the Letters of the Alphabet and Method of Teaching Deaf-Mutes to Speak [3].

1.3.3 Algerian Sign language

Algerian Sign Language is the visual-spatial language used by the Deaf community in Algeria ,It has its own distinct vocabulary, grammar, and cultural expressions, reflecting the rich linguistic and cultural heritage of the Algerian Deaf community.

1.3.3.1 Definition

«Langue des Signes Algérienne (LSA/asl), also known as Algerian Sign Language is a descending form of Old French Sign Language (LSF). Various dialects of LSA, including Laghouat Sign Language, Adrar Sign Language, Algerian Jewish Sign Language (also known as Ghardaia Sign Language), and many others, are spoken in Algeria because different Algerian Sign Languages are being used as many Deaf communities are in Algeria».[27]

The LSA [28] was officially recognized by the Algerian law on the protection and promotion of persons with a disability enacted on May 8, 2002.

1.3.3.2 LSA Datasets

The only ASL dataset found was published in May 2023 named Alabib-65 [29], and it is no open-source dataset.

1.3.4 American Sign Language

Previously we mentioned The absence of an open-source Algerian sign language dataset. Due to this deficiency, we chose the American sign language described above.

1.3.4.1 Definition

«American Sign Language (ASL) is a complete, natural language with the same linguistic properties as spoken languages, with grammar that differs from English. ASL is expressed by movements of the hands and face as It is shown in Figure 1.5. It is the primary language of many North Americans who are deaf and hard of hearing and is also used by some hearing people» [30].

Figure 1.5: A young boy signs "I love you" in ASL [4].

.

1.4 Dialects

Dialects are regional or social variations of a language that differ in pronunciation, vocabulary, and grammar. They reflect the diverse linguistic and cultural characteristics within a specific language, providing unique insights into local identities and traditions

1.4.1 Definition

A dialect is «the form of a language that is spoken in one area with grammar, words, and pronunciation that may be different from other forms of the same language» [31]. Exemple : arabic is the official language of Algeria ,tunisia and moroco while each one of them has its own dialect .

The lack of grammatical norms in dialects presents a challenge for the NLP field.[32] For instance, in Algeria, whenever you travel to the east, you will hear Tunisian dialects being spoken. The same is true for the west of the nation, where people speak a more complicated kind of Moroccan dialect.

Figure 1.1 shows examples of some words in three dialects and Modern Standard Arabic [4] .

Modern Standard Arabic(MSA)	Algerian Dialect	Tunisien Dialect	Morocan Dialect
اجلس	اقعد	اقعدي	اقلس
كثير	بزاف	برشا	بزاف
ماذا	وشنو	شنوا	شنو

Table 1.1: Comparison between MSA,Algerian,Moroccan and Tunisian dialect.

4

Modern Standard Arabic Generally referred as MSA (Alfus'ha in Arabic), is the variety of Arabic which was retained as the official language in all Arab countries, and as a common language. It is essentially a modern variant of classical Arabic. Standard Arabic is not acquired as a mother tongue, but rather it is learned as a second language at school and through exposure to formal broadcast programs (such as the daily news), religious practice, and newspaper [5].

1.4.2 Dialects in Arab countries

The majority of Arab nations ,[5] have Arabic as their official language and it is taught in schools and used in official speeches, news articles, and governmental administration. Parallel to this, Arabic-speaking individuals utilize their own dialects, which vary from region to country and are derived from Standard Arabic for informal conversations, songs, and films.

Disglossia is a linguistic phenomena that occurs when two different languages are spoken by members of the same speech community. It is noticed Standard Arabic extensively written but rarely used in common conversation in all Arab countries, while dialect is frequently spoken but hardly ever written. Thus, written Arabic has been the subject of numerous works in the NLP field. The study of Arabic dialects, in contrast, lagged behind at the time Interest for them is recent.

For Middle Eastern dialects,first research only recently started. The study of Maghrebian dialects is barely getting started. Dialects are under-resourced languages that lack NLP resources despite being used far more than written Arabic.

1.4.3 Algerian Dialect

Low-resource languages [33] such as African and Arabic dialects received less attention due to the lack of data and their specific complex morphology. Algerian dialect where some words from it are presented in Figure 1.6 which is one of the low-resource dialect is spoken by 44 Million people but lacks publicly available datasets. Indeed, MSA is the most common written language in official documents, books, and newspapers in Algeria. However, the local dialect is very frequent in informal communications, messaging, or the social media sphere.

A recent study [34] showed that 74.6% of the Algerian web-generated content (mostly on Facebook) is conveyed in dialectal Arabic rather than MSA and 62% of this content is transcribed in Roman alphabet characters (which is also known as Arabizi).

1.4.4 ALgerian Dialect Specification

Our need to translate the text into the Algerian dialect prompted us to examine each aspect of its text structure, including its syntactic, semantic, and pragmatic components.

Figure 1.6: Some words from the Algerian Dialect .

We learned [5] that spoken languages, such as Algerian dialects, do not always follow the rules, which is what makes them challenging to work with. Additionally, dialects in Algeria vary by location and wilaya. In our case, we used northern Algerian dialect (mostly spoken in Algiers and the cities around), and we've highlighted some of its most distinguishing characteristics in comparison to MSA above.

1.4.4.1 Vocabulary

The vocabulary of Algerian contains Arabic roots, but the original words have undergone significant phonetic changes, Berber substrates, and numerous new terms and loanwords from French, Turkish, and Spanish [5].
Despite the fact that the majority of this vocabulary is derived from MSA , [5] table 1.2 demonstrates some distinguishing changes in the vocalization in most instances and the

removal or mutation of some letters in other instances (mainly the Hamza [5],

English word	The word in arabic	The word in Algerian Dialect
water	الماء	الما
came	جاء	جا

Table 1.2: Examples to illustrate the removal of el Hamza

1.4.4.2 Verbs

Some ALG verbs, [5] such as "to name" and "to salute," can totally adopt the MSA verb cheme by adhering to the same vocalization. Other verbs are pronounced differently from their respective MSA verbs by employing distinctive diacritics markings, such as the instance of the verbs (to drink). One can produce a different set of dialect verbs by omitting or changing a few letters. We present examples for each of the cases listed in tab 1.7.

Also , due to european colonialism in Algeria, several verbs are derived from other languages, mainly French. The table 1.3 displays a few examples.

Moreover,we note that the algerian verbs can be conjugated in the present ,future and past tense; The table 1.4 shows an example .

ALG Verb	Corresponding MSA Verb	Meaning Situation	Situation
سَلَّمْ	سَلَّمَ	To salute	Same scheme
قَابَلْ	قَابَلَ	To confront	Same diacritics marks
شْرَبْ	شَرِبَ	To drink	Same scheme
كْتَبْ	كَتَبَ	To write	/Different Diacritics marks
جَا	جَاء	To come	
بْقَى	بَقِيَ	To remain	Letters omission
كْلَا	أَكَلَ	To eat	or modification
كَمَّلْ	أَكْمَلَ	To finish	

Figure 1.7: verbs scheme differences between ALG and MSA[5].

5

the Hamza [5] The Hamza is a letter in the Arabic alphabet, representing the glottal stop.

English word	French word	Algerian word
to load	charger	شارجا
to pass	passer	باسا

Table 1.3: Some ALG verbs derived from french [5] .

conjugation for the verb (to eat)	ALG verb conjugation
he is eating	هو راه ياكل
he ate	هو كلا
he will eat	هو راح ياكل

Table 1.4: ALG verb conjugation [5] .

1.4.4.3 Nouns

There are a variety of nouns [5] when it comes to the Algerian dialect . There are some nouns that are divided from verbs like in the MSA shows in Figure 1.8 and nouns that are originally from the French language shows in Figure 1.5.

Verb	Verbal name	Acive participle	Passive participle
باع	بيع	بايع	مبيوع
To sale	Sale	Seller	Sold

Figure 1.8: Example of ALG nouns derived from a verb [5].

English word	French word	Algerian word
the sidewalk	le trottoire	الطروطوار
the garden	le jardin	الجردينة
the transportation	le transport	الطرونسبور

Table 1.5: Some algerian nouns which do have a relation with french nouns.

1.4.4.4 Syntactic level

A declarative sentence in ALG can be written in almost any word arrangement as it is shown in Figure 1.9.[5] ALG phrases may, in fact, start with the verb, the subject, or even the object in frequent usage. This ranking is based on how significant the speaker believes each of these things to be.

Order	Dialect Sentence	English
SVO	الولد راح للمسيد	The boy went to school
VSO	راح الولد للمسيد	
OVS	للمسيد الولد راح	
OSV	الولد للمسيد راح	

Figure 1.9: Example of word order in a ALG declarative sentence [5].

While any sentence in Algiers can be interpreted as a question in one of the following ways:

- It could be said in an accusatory tone of voice, such as : راح تقرى ؟ (Will you revise?).

-By using a pronoun or particle of interrogation ,such as : وين راح تقرى؟ (where will you revise?).

Analyzing the Negative form ,in most cases the particles and are employed to represent negation.ما is used in both the MSA and the Algiers dialect, however the way in which it is negated varies between the two languages, whereas ماشي is unique to the ALG. The negative form is obtained in ALG in a variety of ways using these particles [5].we provide several examples in figure Figure 1.10 labeled with each situation .

Case	ALG	MSA	English
1	لعبت ما لعبتش	لعبت لم تلعب / ما لعبت	she played she didnt play
2	راهي مريضة ما راهيش مريضة	إنها مريضة ليست مريضة	She is ill She is not ill
3	هوما كتبو ماشي هوما كتبو	هم كتبوا ليسوا هم من كتبوا	They wrote They are not those who wrote
4	هوما كتبو ماشي هوما آلي كتبو	هم كتبوا ليسوا هم الذين كتبوا	They wrote They are not those who wrote
5	الولد مريض ماشي مريض الولد	الولد مريض ليس الولد بمريض	The boy is ill The boy is not ill
6	الولد مريض الولد ماشي مريض	الولد مريض الولد ليس مريضا	The boy is ill The boy is not ill

Figure 1.10: Declarative sentences with their Negation.[5].

1.5 Natural Language Processing

Natural Language Processing (NLP) is a tract of Artificial Intelligence and Linguistics, devoted to making computers understand the statements or words written in human languages.[35] It came into existence to ease the user's work and to satisfy the wish to communicate with the computer in natural language, and can be classified into two parts i.e. Natural Language Understanding(NLU) or Linguistics and Natural Language Generation (NLG) which evolves the task to understand and generate the text. Noah Chomsky, one of the first linguists of the twelfth century that started syntactic theories, marked a unique position in the field of theoretical linguistics because he revolutionized the area of syntax .

1.5.1 NLP history

development and research have evolved over time. Here is a brief overview of the key milestones:

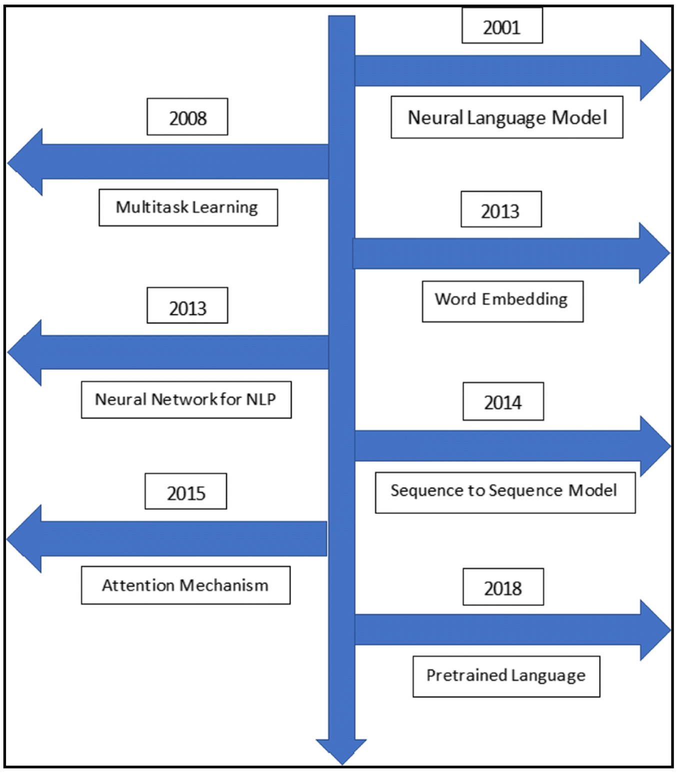


Figure 1.11: A walkthrough of recent developments in NLP [6]

In the early stages of natural language processing (NLP), linguists manually defined grammatical rules to process and analyze language using rule-based systems. However, these systems had limitations in handling complex language patterns. The advent of statistical models and machine learning techniques revolutionized NLP, enabling computers to learn patterns from large amounts of text data. Recent developments in deep learning, facilitated by the availability of massive datasets and computational power, have further advanced NLP, leading to breakthroughs in tasks such as machine translation, sentiment analysis, and question-answering systems.

1.5.2 Components of NLP

NLP can be broadly classified into two main components: Natural Language Understanding (NLU) and Natural Language Generation (NLG). These components focus on different aspects of language processing and play crucial roles in NLP applications.

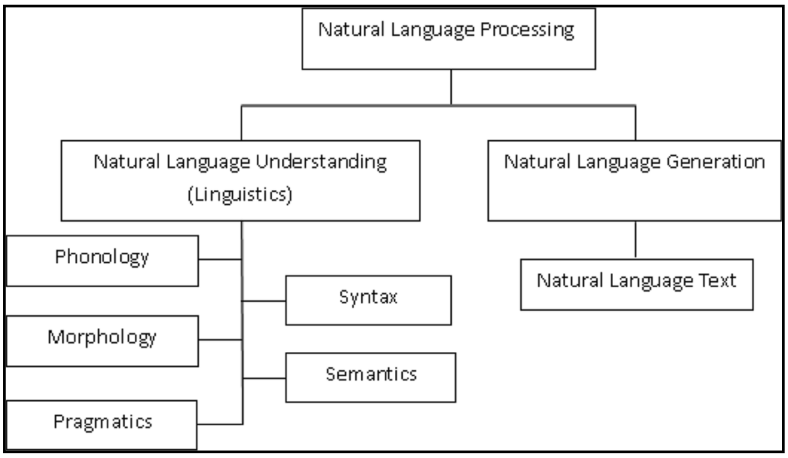


Figure 1.12: broad classification of NLP [6]

1.5.3 Common NLP tasks and techniques

NLP is a crucial field within computer science that extensively utilizes machine learning and computational linguistics. The primary goal of NLP is to facilitate and enhance the interaction between humans and computers. By learning the syntax and meaning of human language, machines can process it and provide output to the user.

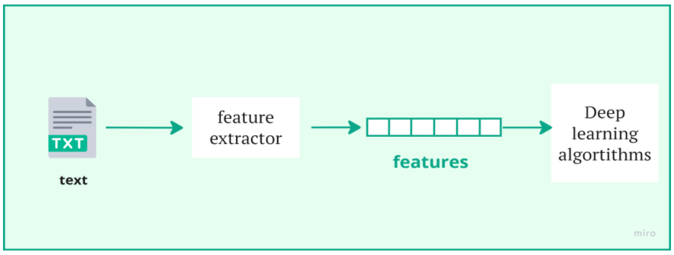


Figure 1.13: feature extraction

1.5.4 NLP use cases

NLP has found innovative applications in various fields, including sign language detection using object recognition and enables automatic language translation through machine translation tools, enhancing worldwide communication. With sentiment analysis, NLP also allows us to assess feelings and views from text, giving useful information for businesses and other application as shown in Figure 1.14.

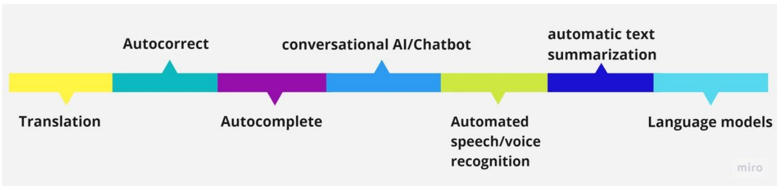


Figure 1.14: deferent uses of NLP

1.5.5 Machine translation and NLP

Machine translation (MT) is a field in computational linguistics that focuses on using software to translate text or speech from one language to another.[36] While the basic approach involves word substitution, it doesn't guarantee high-quality translations.

To address the challenge of recognizing multiple phrases and improving translation accuracy, more advanced methods have emerged, particularly statistical and neural techniques.

These techniques leverage sophisticated algorithms and models to enhance translation quality. In this automated process, human involvement is eliminated, and the entire conversion process is carried out by the machine. There are three primary types of machine translation systems: rule-based, statistical, and neural, each employing different approaches and algorithms to achieve translation capabilities.

Since 2013, internet search giants like Google and Microsoft have started investigating the use of machine neural networks, which have led to significant improvements in the quality of machine translation.

1.5.5.1 Neural machine translation

The neural machine translation system architecture [7] consists of two core parts: an encoder and a decoder which are implemented using recurrent neural networks (RNNs). Figure 1.15 displays the neural network parts used for machine translation. Here, we cover the fundamental structure of the encoder-decoder architecture, which serves as the foundation for the majority of sequence-to-sequence models.

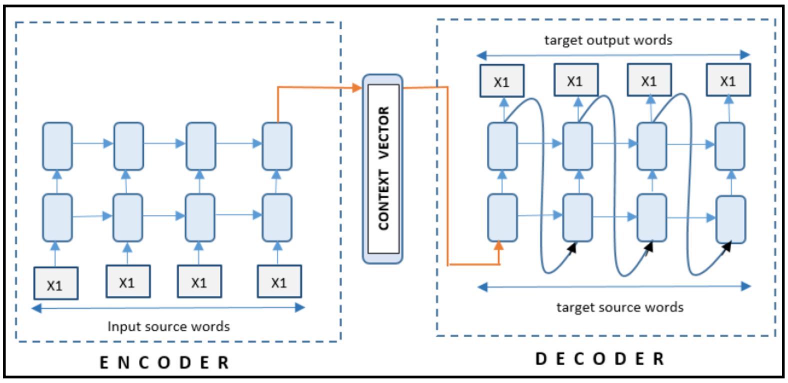


Figure 1.15: Basic encoder–decoder framework.[7]

1.6 Deep learning techniques

Actually, deep learning algorithms, particularly neural networks, have shown remarkable capabilities in understanding and processing complex natural language patterns NLP models can now learn directly from raw text data without the need for extensive manual feature engineering using Deep learning architectures like recurrent neural networks (RNNs), Long short term memory (LSTM) , convolutional neural networks (CNNs), and transformer models, such as the popular BERT (Bidirectional Encoder Representations from Transformers).

1.6.1 Neural networks

A neural network, also known as an NN,[9] The information flow in a neural network is described using the example of the optic nerves in human visual processing. Similar to how the optic nerves receive visual input, the first layer gets the raw input data. Just as neurons farther from the visual nerve get signals from those closer to it, each succeeding layer receives the output from the preceding layer. The last layer of the neural network creates the system's output.

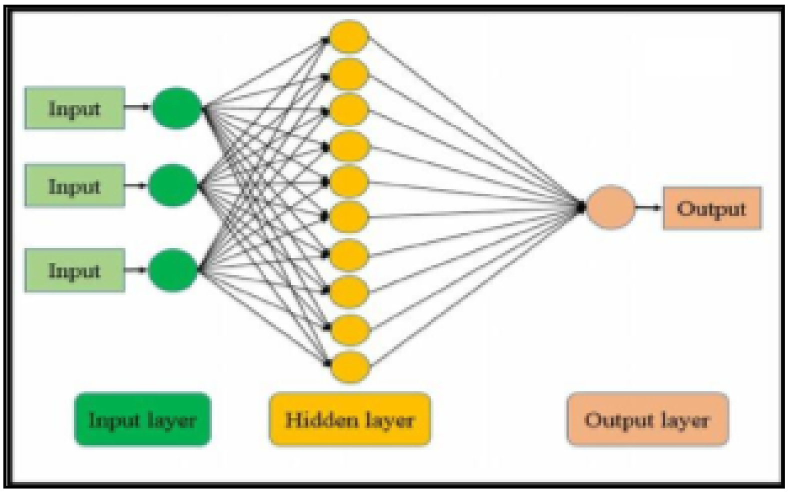


Figure 1.16: Deep neural network.[8]

1.6.2 Convolutional Neural Networks

convolutional neural network (CNN) [10] is a type of deep learning neural network that is generally used to analyze visual imagery. CNNs are similar to regular neural networks in that they are made up of neurons that have learnable weights and biases. However, CNNs have a unique architecture that is specifically designed to take advantage of the 2D structure of images. One of the main components of a CNN is the convolutional layer. This layer is made up of a set of filters (also called kernels) that are used to detect certain features in an image.

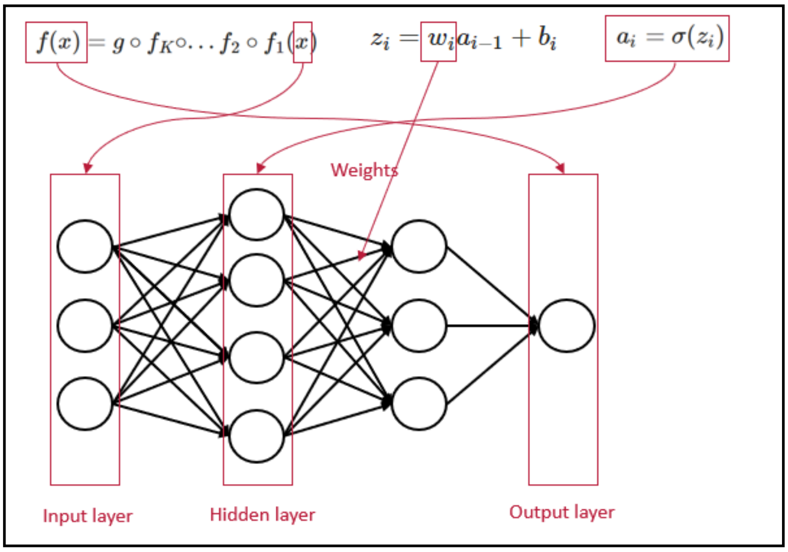


Figure 1.17: how deep neural network work.[9]

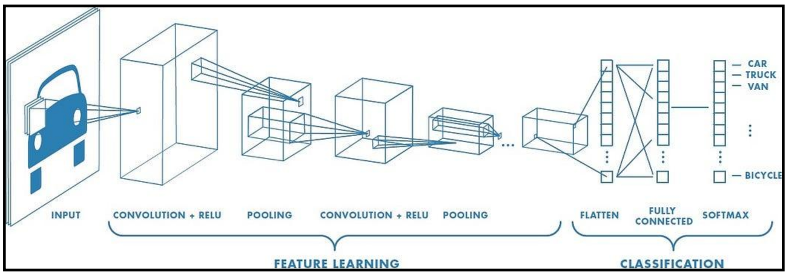


Figure 1.18: Neural network with many convolutional layers sourse.[10]

1.6.3 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) [11] are a crucial variation of neural networks extensively employed in Natural Language Processing. In typical neural networks, inputs undergo processing through layers to generate outputs, assuming that consecutive inputs are independent. However, this assumption does not hold true in many real-life situations. For instance, when predicting stock prices at a given time or forecasting the next word in a sequence, the dependency on previous observations becomes essential.

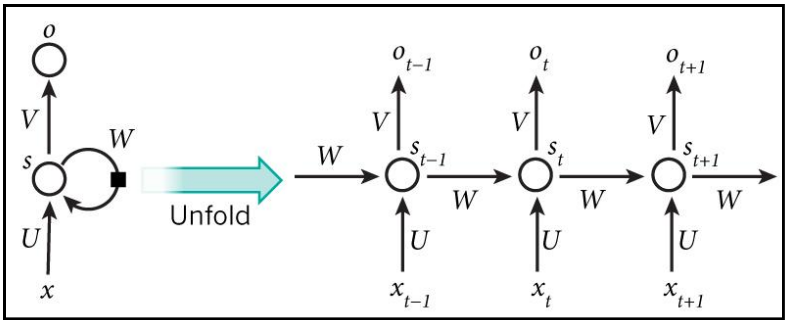


Figure 1.19: A recurrent neural network and the unfolding in time of the computation involved in its forward computation source[11].

1.7 Related works

No work was found for the translation from sign language to Algerian dialect, or vice versa. For Algerian sign language detection, an alabib-65 Algerian sign language dataset [29] was published last June. While the dataset is not open source, we opted for an ASL dataset as there are a lot of different ones and documentation available, A list of these datasets are shown in Table 2.5 . For the Algerian dialect previous works, we found the Dziribert model [33], which does sentimental analysis, while for us, our goal is translation.

DataSet	Utility
American Sign Language Dataset [16]	Image detection
ASL Alphabet [17]	Image detection
WASL (world level American Sign Language) [18]	video detection
Sign Language MINST [19]	Image detection

Table 1.6: Some Dataset Founded with there utilities

1.8 Conclusion

This chapter explained why we targeted deafness and hearing loss individuals in our work, sign language history, and its reality. Also, we explained dialects in general, arab countries' dialects, and precisely the Algerian dialect uniqueness. we set the stage for the next chapter where we delve into the technical aspects of our modeling approach.

Chapter 2

Modeling Chapter

2.1 Introduction

In this chapter, we will introduce the architecture of our proposed model. We will provide a broad model overview and then delve deeper into its components. We will discuss their origins and how we have adapted them to address our goals and challenges. Additionally, we will describe the different parts of our system, the details of each phase, and their interactions, which will be presented in the following subsections.

2.2 System Overview

Our work aims to develop a web application that allows translating American Sign Language (ASL) into the Algerian dialect that Algerian-hearing individuals can use to understand deaf persons who use ASL. To develop a suitable solution for a system, it is necessary to rely on a technique once the system has been defined and the fundamental principles have been established. The chosen methodology should be tailored to the specific problem at hand and facilitate the solution's design process.

So we propose two systems:

- Detection and extraction of text from signs in real-time.
- Translation from English to the Algerian dialect.

To provide a clear and organized overview of the process, the following Figure 2.1 presents the steps involved in achieving our goal.

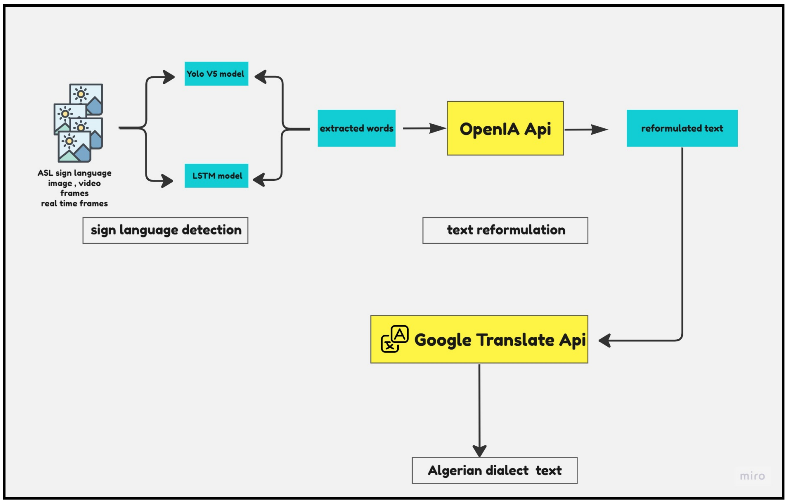


Figure 2.1: Sign Language to Algerian Dialect translation system overview

The Figure represents the architecture of our proposed system, which is divided into three main approaches.

The first approach is focused on sign detection. In this stage, we collect data to train our different models: the LSTM model and the YOLOv5 model. Once these models perform sign detection, their outputs become the inputs for our next approach.

The second approach involves text reformulation using the OpenAI API. This step aims to take the results from sign detection and reformulate them into a coherent sentence. This reformulated text then serves as the input for our next approach.

The third approach deals with automatic translation into the Algerian dialect. The reformulated textual data undergoes an automatic translation process, converting it into Algerian dialect

In the overall architecture, data is collected and utilized to train the sign detection models. The results from this detection process are then used for text reformulation, which is subsequently translated into the Algerian dialect.

This architecture follows a logical and linear sequence, where each approach utilizes the outcomes of the preceding step to provide meaningful input for the subsequent stage.

2.2.1 Detection and extraction of text from signs in real-time

We tested two models to extract text from signs during our study :

- YOLOV5 model.
- LSTM model.

The following Figure2.2 explained the process.

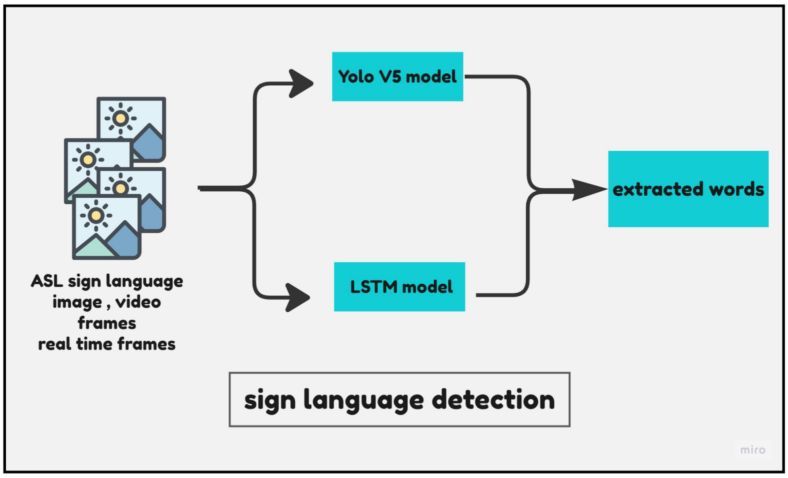


Figure 2.2: A system overview of the sign detection step

2.2.1.1 Data Collection

We created two datasets to train our Sign detection model, one for the LSTM model using media pipe and the other for the YOLOV5 model, as the requirements for each model differ.

Data collection for LSTM model

MediaPipe [37] is a framework that allows developers to create cross-platform multi-modal (video, audio, and time series data) ML pipelines. It contains various human body identification and tracking algorithms trained on Google's massive and diverse dataset. They track crucial locations on multiple body sections as the skeleton of nodes and edges or landmarks.

Moreover, Hand tracking solution,[38] utilizes a machine learning pipeline at the backend consisting of two models that work in coordination:

- Palm Detection Model.
- Landmark Model.

The palm detection model provides an accurately cropped palm image, which is then passed along to the landmark model for further processing.[38] This process reduces the use of data augmentation (e.g., rotations, flipping, and scaling) in deep learning models and focuses more on landmark localization.

Traditionally, the hand is detected in the frame and then landmark localization is performed. It is important to note that using the ML pipeline presents significant challenges with a different strategy in this Palm Detector.

Hand detection is a complex process that requires image processing and thresholding as well as handling a variety of hand sizes, which consumes a significant amount of time.

In order to detect hands, the palm detector must be trained first, which estimates the bounding boxes around rigid objects such as palms and fists, making it easier to detect hands with coupled fingers rather than detecting hands directly.
Second, an encoder-decoder serves as an extractor for a larger scene context.

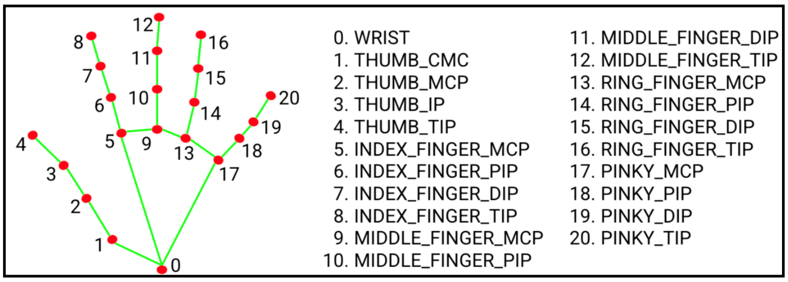


Figure 2.3: mediapipe handmark.[12]

Data collection for yolov5 model

During the training process for ASL recognition, gathering a wide range of data comprising both photos and video frames is crucial. This diverse dataset enables the model to acquire knowledge and proficiency in recognizing signs under various circumstances, encompassing varying camera angles and lighting conditions. By obtaining multiple instances for each sign from different viewpoints, the model can enhance its ability to generalize and comprehend the nuances of sign variations as they manifest in real-life situations. As a result, the model's accuracy and resilience in recognizing ASL signs are significantly enhanced.

2.2.1.2 Models

We work with two different models :

YOLOv5 model

Our experiment carried us to use the deep learning-based architecture YOLOv5.[1].It is a model in the You Only Look Once (YOLO) family of computer vision modelsand [39]it origine came from the YOLOv3 PyTorch.[2] repository by Glenn Jocher.

The Figure 2.4 bellow shows the architecture of YOLOv5 ,where [13] it consiste of three parts:

- Backbone: CSPDarknet.
- Neck: PANet.
-Head: Yolo Layer.

Data [13] is first inputed into CSPDarknet for feature extraction and then transferred to PANet for feature fusion. Finally, the Yolo layer returns recognition results (class, score, location, size) as it is shown in Figure 2.4 .

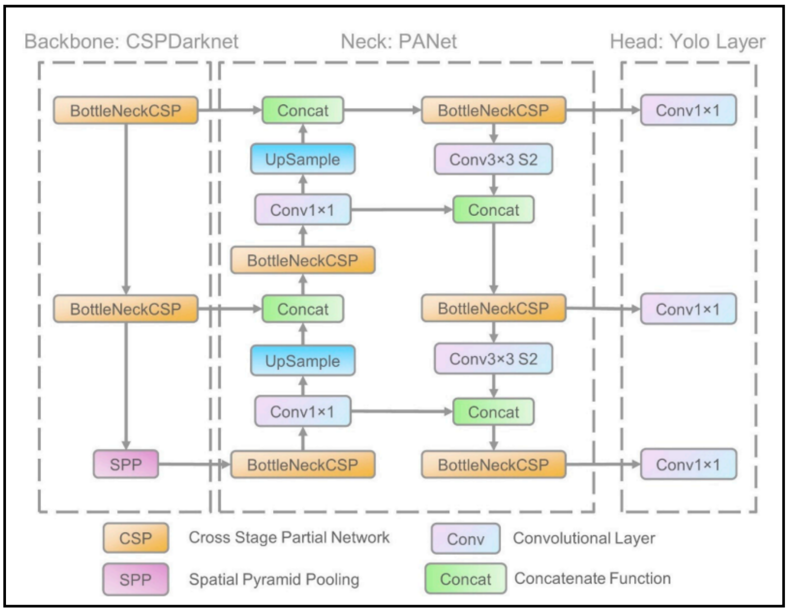


Figure 2.4: The network architecture of Yolov5. [13]

[1]https://github.com/ultralytics/yolov5
[2]https://pytorch.org/

YOLOv5 is lightweight and quick. Use less computing resources than other current state-of-the-art architectural models while maintaining detection accuracy close to current state-of-the-art detection models, and it is far faster than the different YOLO variants. Moreover, it extracts the feature map from the picture using CSPNET [40]. It also uses the Path Aggregation Network (PANet) [41] to improve information flow.

LSTM model

Recurrent neural networks (RNNs) called Long Short-Term Memory (LSTM) networks [42] uses a special gating mechanism to control access to memory cells. This gating mechanism enables it to successfully retain and apply information across longer sequences.

Specialized units in this architecture [43] are referred to as memory blocks in the recurrent hidden layer. The temporal state of the network is stored in these memory blocks, which are made up of cells with self-connections. They also have special multiplicative units known as gates that control the information flow. Each memory block originally featured an input gate and an output gate. The output gate governs the outflow of cell activations to the rest of the network, while the input gate regulates the flow of input activations into the memory cell. A forget gate was then added to solve a flaw in LSTM models.

They can handle [44] continuous input streams thanks to the LSTM architecture, which eliminates the need for subsequence segmentation. Before incorporating it as input through the self-recurrent link, the forget gate scales the cell's internal state. The cell can selectively forget or reset its memories thanks to this adaptive process. Additionally, current LSTM structures have peephole connections that link internal cells to their respective gates, allowing for exact timing of LSTM, RNN systems. understanding the results.
the Figure 2.5 above shows a representation of the LSTM architecture .

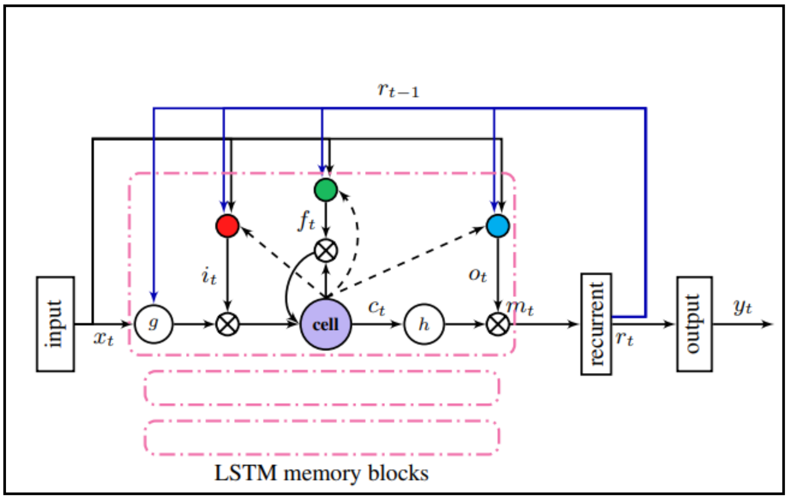


Figure 2.5: LSTM architecture. [14]

2.2.1.3 Model evaluation

Model evaluation is a crucial step in assessing the performance of models. It involves measuring how well the model performs in solving a specific problem. in our case we use Two common evaluation metrics for classification tasks are recall and precision

- **Recall**: also known as sensitivity or true positive rate, calculates the ratio of true positives to the sum of true positives and false negatives. to calculate recall we apply the following formulas:

  Recall = True Positives / (True Positives + False Negatives)

- **Precision**: Precision focuses on minimizing false positives, ensuring that the model doesn't falsely classify negative instances as positive it calculates the ratio of true positives to the sum of true positives and false positives. To calculate precision we apply the following formulas:

  Precision = True Positives / (True Positives + False Positives)

2.2.2 Text reformulation

NLP encompasses various techniques and tasks related to understanding and generating human language using computational methods, and text reformulation is considered an NLP task. Text reformulation refers explicitly to paraphrasing or rephrasing a given text while preserving its original meaning. This task often involves techniques such as word substitution, sentence restructuring, and grammar modifications to produce alternative expressions with similar semantics. Many NLP models and algorithms were employed to automate text reformulation, such as Seq2Seq Models [45], Word2Vec models [46] Text-to-Text Transfer Transformer (T5) [47], and Neural Machine Translation [48]. We chose to work with a seq2seq model as it has given an accurate result and one of the large pre-trained models (GPT 3.5) work with.

2.2.2.1 Seq2seq model

Deep Neural Networks (DNNs), [15] struggle with sequences because they need the inputs and outputs dimensions to be known and stable. Sequence-to-sequence models were proposed as a result. A huge fixed dimensional vector representation is obtained by reading the input sequence through one timestep at a time using one LSTM, and the output sequence is then extracted from that vector using a second LSTM, as illustrated in Figure 2.6. In essence, the second LSTM is a recurrent neural network language model, but it depends on the input sequence.

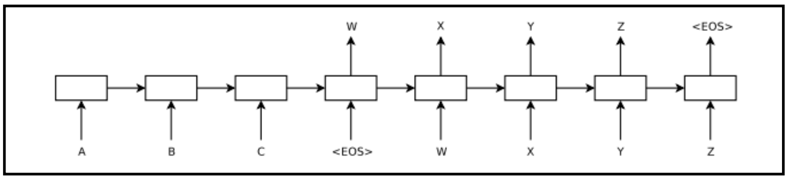


Figure 2.6: sequence to sequence architecture [15].

2.2.2.2 Generative Pre Training of a language model

GPT [49] is an autoregressive language model that employs deep learning to generate text that resembles human speech and was released by OpenAI in 2020. It will produce text that answers questions when provided a cue. In the first version of GPT a large dataset was passed to the model to generative pretrained and then it was fine-tunned to another task like :classification,question answering ,language inference ,sementic symilarity).

GPT2 [50] removed the fine tuning step and has moreparameters than the GPT 1 .While GPT 1 was fine tuning for each NLP task GPT 2 can be used directly for any NLP task .Fine tuning was a big drawback for GPT 1 because fine tuning is not easy and take many time and one fine tuning model can not be used for other nlp task.

2.2.2.3 Signy.io text reformulation pipeline

wWe have noticed that our sign detection model has repetitive words due to the model precision when we have to repeat a sign several times to be detected, as shown in Figure 2.8. To solve this problem and not impact the translation result, we made a reformulation pipeline according to our problem, as highlighted in blue in Figure 2.7 :

- If the sign detection model output contains repetitive words, we remove them.
- If no repetitive word is detected, we pass to the text reformulation step with GPT.

-As an output, we will get a reformulated, coherent sentence ready to be translated.

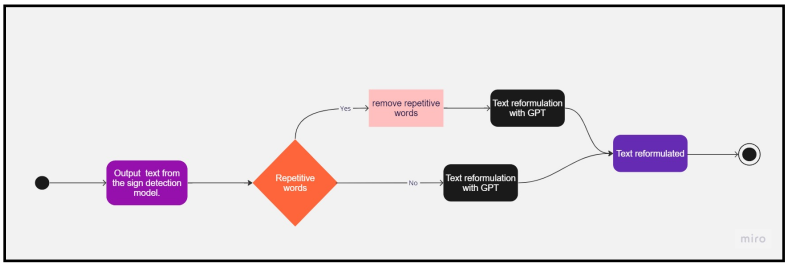


Figure 2.7: Text reformulation procedure

Figure 2.8: Text reformulation procedure

2.2.3 Machine translation with google translate

Google's Neural Machine Translation System,[16] which uses the conventional sequence-to-sequence learning framework [15] with attention [49] , was presented in 2016. An encoder network, a decoder network, and an attention network make up its three parts. A source sentence is converted by the encoder into a list of vectors, one vector for each input symbol. The decoder generates one symbol at a time from this list of vectors until the special end-of-sentence symbol (EOS) is generated. A connection between the encoder and the decoder's attention module enables the decoder to concentrate on various parts of the source phrase while decoding. the Figure 2.9 above shows Google's neural network architecture.

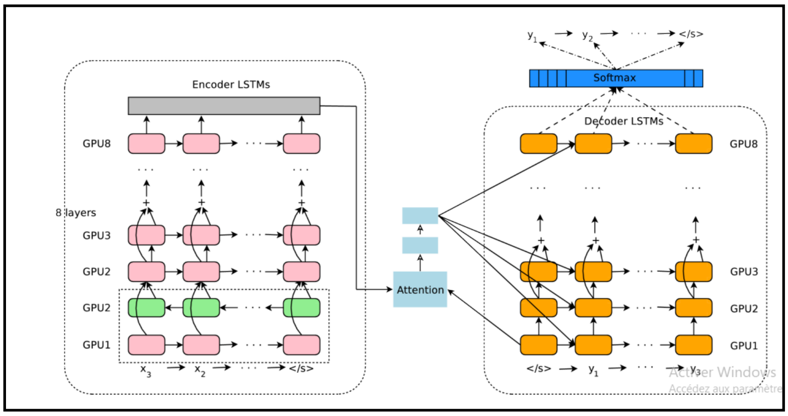


Figure 2.9: Google neural network architecture. [16]

2.2.4 Word-for-word translation

Literal translation,[51] also known as direct translation or word-for-word translation, entails translating each individual word in a document without considering how those words are employed in a phrase or sentence. Iterative translation causes idioms to be mistranslated, which is a major issue for machine translation.

2.2.5 Signy.io machine translation pipeline

Our main goal was to translate into the Algerian dialect. However, as it is a low-resource language, as mentioned in Chapter One, we cannot go directly from a rich language like our case study into a low-resources one, so we opted for Arabic as a pivot language because the Algerian dialect extracts its main feature from the MSA. To achieve the translation tasks, we passed the following steps shown in figure 2.10:

- translate the output text of the reformulation step from English to Arabic.

-Save the length of the Arabic sentence.

- Search each word in Arabic to the Algerian dialect dataset.

-if the word is found, replace it with its corresponding meaning in the Algerian dialect; if not, keep it as it is.

-iIf the length of i" is equal to the length of the sentence, stop translating if not, go to the next word.

Our dataset contained a set of nouns with their plural, verbs with their conjugation in the present, past, and future, and a set of verbs and pronouns as shown in figure 2.11.

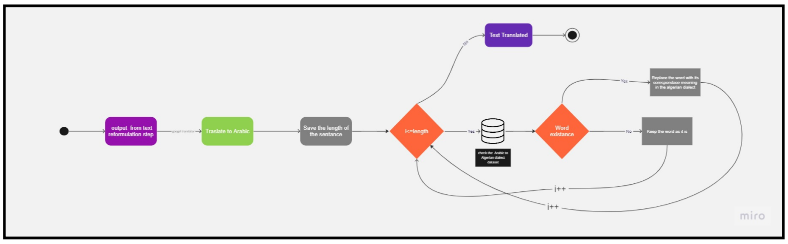


Figure 2.10: Signy.io translation pipeline

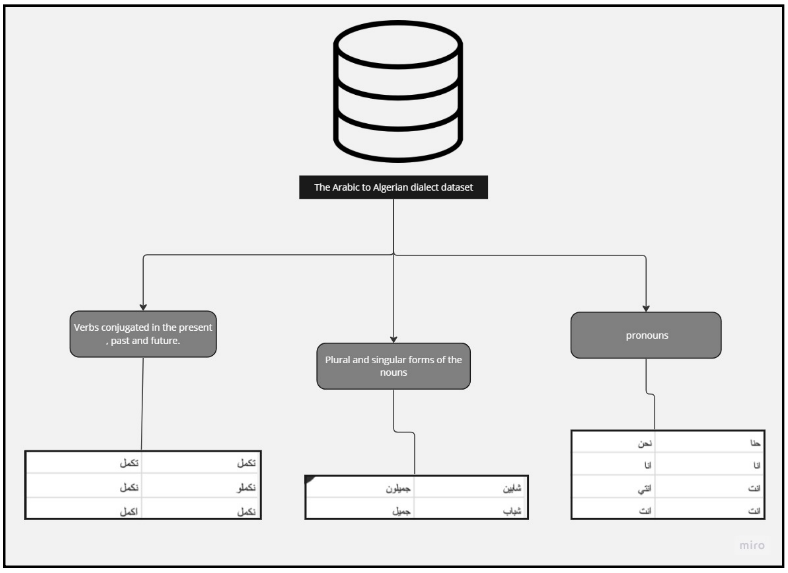


Figure 2.11: Arabic to Algerian dialect dataset content

2.3 Signy.io platform diagrams

To test our solution we opted for a web application and the subsections below describe our application use cases and UML diagrame.

2.3.1 Use case diagram

Figure 2.1 shows a use case diagrame of the signy.io platfor .

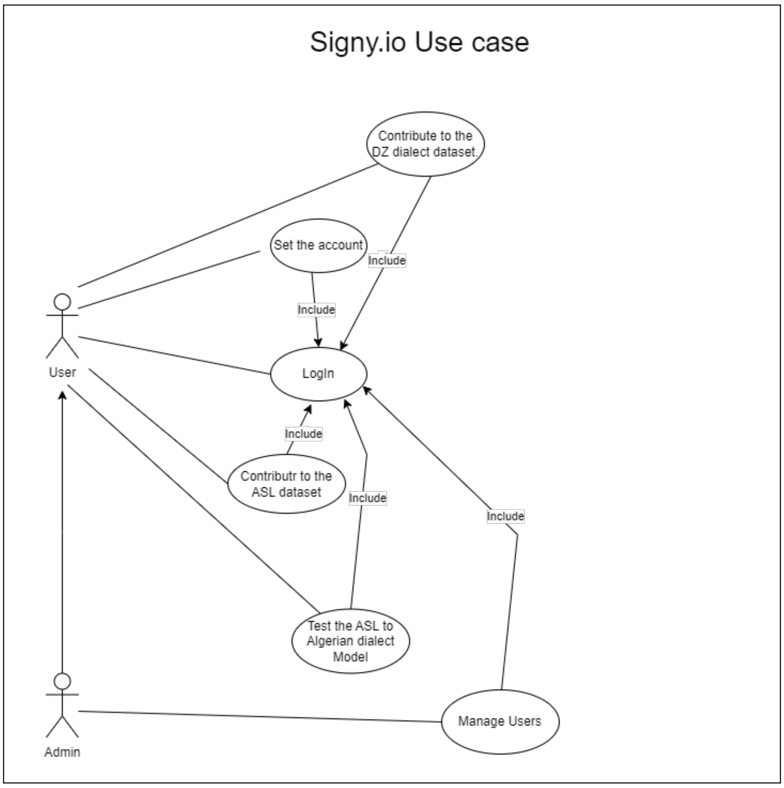


Figure 2.12: Signy.io platform use case

The tables 2.1 ,2.2, 2.3,2.4 and 2.6 shows respectively descriptions of the login use case, set account use case, contribute to the DZ dialect dataset use case, contribute to the ASL dataset use case.

Use Case	Login
Actor	User,Admin
Goal	Acess to signy.io platform
Description	To access to the platform the user should inset his email and password
Pre-condition	The email and password should be equal to one of the columns saved in the dataset
post-condition	The user can access the platform and see different pages he has access to them
Exception	If the user do no thave an account he will not be able to access the home page, and an error message will appear

Table 2.1: description of login use case

Use Case	Set account
Actor	User,Admin
Goal	Update user information
Description	To access to the platform the user should inset his email and password
Pre-condition	The email and password should be equal to one of the data saved in the database
post-condition	The user can access the platform and see different pages he has access to them
Exception	If the user do no thave an account he will not be able to access the page, and an error message will appear

Table 2.2: description of Set account use case

Use Case	Contribute to the DZ dialect dataset
Actor	User,Admin
Goal	Contributing to the dataset
Description	To access to DZ dialect dataset the user should log in
Pre-condition	The email and password should be equal to one of the data saved in the database
post-condition	The user can add words and sentences to the dataset by filling out the form
Exception	If the user do not have an account he will not be able to access the page, and an error message will appear

Table 2.3: description of Contribute to the DZ dialect dataset use case.

Use Case	Contributr to the ASL dataset
Actor	User,Admin
Goal	Contributing to the dataset
Description	To access to ASL dataset contribution page, the user should log in
Pre-condition	The email and password should be equal to one of the data saved in the database
post-condition	The user can add sign language videos to the dataset by filling out the form
Exception	If the user do not have an account he will not be able to access the page, and an error message will appear

Table 2.4: description of contribute to the ASL dataset use case

Use Case	Test the ASL to Algerian dialect Model
Actor	User,Admin
Goal	test the model
Description	To test the model, the user should log in
Pre-condition	The email and password should be equal to one of the data saved in the database
post-condition	The user can test the model
Exception	If the user do not have an account he will not be able to access the page, and an error message will appear

Table 2.5: description of contribute to the ASL dataset use case

Use Case	Manage Users
Actor	Admin
Goal	update,delete and see users data
Description	To manage the users, the user should log in and have the role of an admin
Pre-condition	The email and password should be equal to the admin data saved in the database
post-condition	The user can manage users
Exception	If the user do not have an account he will not be able to access the page, and an error message will appear

Table 2.6: description of manage users dataset use case

2.3.2 UML diagrame

Figure 2.13 shows a UML diagram describing our platform and table 2.7 describe in detail our UML diagram.

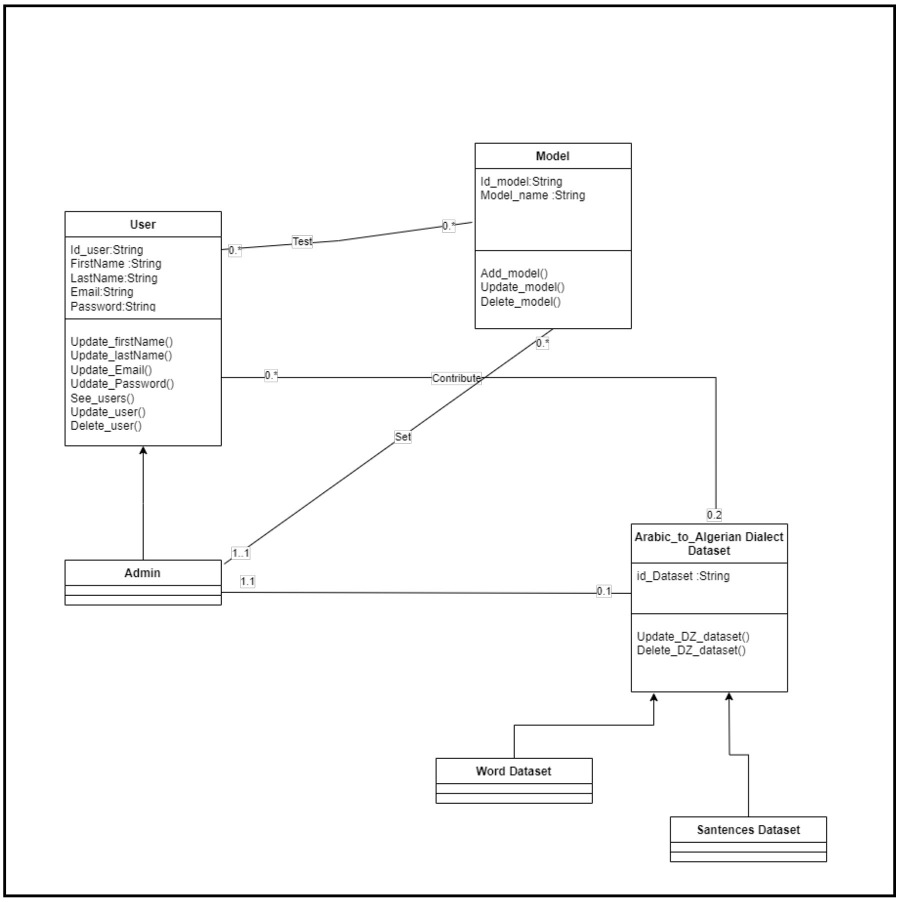


Figure 2.13: Signy.io UML diagram

Class Name	Role	Attribute
User	In this class, we found the user attribute functionalities also, this class is the mother class of the admin class.	Id user,firstname,lastname,email, password
Admin	This class inherits from the user class its attribute and functionality and adds its specificity.	Id user,firstname,lastname,email ,password
Model	This class contains the attribute and functionalities related to the model class.	Id model,name model.
Arabic to Algerian Dialect Dataset	This class contains the attribute and functionalities related to the dataset	Id dataset
word dataset	This class inherits from Arabic to Algerian Dialect Dataset its attribute and functionality and adds its specificity	Id dataset

Table 2.7: description of Signy.io UML diagram

2.4 Conclusion

this chapter has laid the groundwork for our work on detecting American Sign Language and translating the extracted text into Algerian dialect. We have developed a robust architecture, prepared and processed the specific dataset, extracted key features, and trained our models using various techniques. These preliminary steps have prepared us for the test and implementation, which will be addressed in Chapter 3.

Chapter 3

Test and implemechntation

3.1 Introduction

In the previous chapter, we presented a comprehensive architecture for our proposed system. In this chapter, our focus shifts to the crucial test and implementation process of that architecture. To begin, we will introduce the environment, libraries, and APIs employed throughout our process.Moving forward, we will present the results obtained from our model. This includes a thorough analysis of the performance measures Finally, we will conclude by presenting our platform.

3.2 Implementation setup

In This section we will discuss the process, tools, libraries, and development environment specifications used to develop our application.

3.2.1 Work environment

3.2.1.1 Working enviroment for LSTM model

We collected our LSTM model data and trained our model in our local environment using our local resources detailed in Table3.1 .

Operating system	Processor	**Ram**
WINDOWS 10	Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz 2.50 GHz	8 GO

Table 3.1: System Specifications.

3.2.1.2 Google colaboratory

For our YOLOv5 model training and testing and the creation of our final model, we opted for Google Colaboratory .[1]It is a cloud service offered by Google based on Jupyter Notebook. This platform allows us to train Machine Learning models directly in the cloud for free. The added benefit of utilizing Google Colaboratory is its provision of free computing resources, including GPU training acceleration, which significantly enhanced the efficiency of our development process.

3.2.2 Libraries and tools

We chose Python as the programming language for our model's creation because it has gained immense popularity in machine learning and artificial intelligence due to its simplicity, readability, and an extensive collection of libraries and frameworks it has which facilitate our data processing and build our deep learning models.
In the Table 3.3 below, we list the Tools and APIs we used to create our solution and their description.

3.3 Sign language detection

In this stage, we collect data to train our different models: the LSTM model and the YOLOv5 model.

3.3.1 Data collection

To create our dataset and train our Sign detection model, We made two different datasets, one to train the LSTM model using Mediapipe and the other to train the yolov5 model.

3.3.1.1 Landmarks detection using MediaPipe

To train our sign detection model, We had to create our dataset as no open source with the needed criteria was available. We made a dataset of 120 signs with 30 frames to

[1]https://colab.research.google.com/

[1]

Tool and APIs	Description
OpenCV[52]	Open Source Computer Vision Library is an open source computer vision and machine learning software library.
PyTorch [53]	an open source machine learning Python software library based on Torch developed by Meta. PyTorch makes it possible to perform the tensorial calculations necessary in particular for deep learning.
TensorFlow[54]	TensorFlow is an open-source machine learning framework developed by Google. It provides a comprehensive set of tools, libraries, and resources for building and deploying machine learning models.
Media pipe [55]	MediaPipe is a framework that allows developers to create cross-platform multi-modal (video, audio, and time series data) ML pipelines
Open iA[56] API	interface provided by OpenAI enables developers to leverage the power of OpenAI's advanced natural language processing capabilities to generate text...
LabelImg[57]	open-source graphical image annotation tool widely used for object detection tasks. It provides a user-friendly interface that allows you to annotate images by drawing bounding boxes around objects of interest and assigning class labels to those objects.
googletrans[58]	Python library that provides a simple interface for Google Translate API. It allows developers to easily integrate language translation capabilities into their Python applications.
Sklearn[59]	scikit-learn, also known as sklearn, is a popular open-source machine learning library for Python. It provides a wide range of tools and algorithms for various machine learning tasks
Numpy[60]	a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
os[61]	The "os" library (short for operating system) provides a set of functions and methods that allow programmers to interact with the underlying operating system on which their program is running.
Google drive [62]	Google Drive or Google Disk in Canada is a cloud file storage and sharing service launched by the Google company , we used it to stock our dataset.

Table 3.3: Tools and APIs Description

achieve that .Using a media pipe holistic model, we saved the extracted key points for each frame as a string.
The data was collected with the openCV library , as we needed to read the frames from our webcam as it is shown in Figure 3.1. Also, we set the media pipe holistic to draw the key points on our face, hands and pose before saving them because, in our dataset, we save the key points and not the frames.
To achieve the detection, we created two variables, mpholistic and mpdrawing,

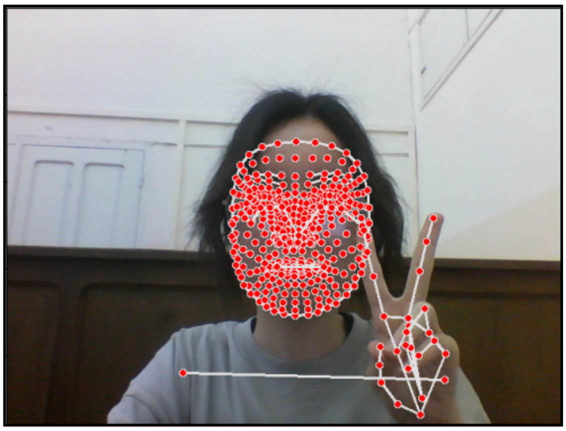


Figure 3.1: webcam capture with keypoints

We Also set a function to convert our frames from BGR to RGB and set it to unwriteable to save memory and then make our detection of signs and flip it again to writable and then from RGB to BGR.
By default, the feed we got from openCV reads the frames in the format of BGR, but when we went to do the detection with mediaPipe, we needed the structure to be in RGB, and we did that transformation using a function from openCV called cvtColor.
Talking about the mediaPipe detection ,it actually makes an initial detection and then from that it will track the keypoints.

we provide a detection confidence score of 0.5 and a confidence tracking score of 0.5 as an input as shown in Figure 3.2. That score presents the likelihood that a detected body part is present in the given frame for the initial confidence score associated with the talking process. The confidence tracking score and the detection confidence score are between 0 and 1, and there is no absolute measure of correctness as the confidence score in machine learning models are typically relative measures that indicate the model level

```
# Set mediapipe model
with mp_holistic.Holistic(min_detection_confidence=0.5, min_tracking_confidence=0.5) as holistic:
```

Figure 3.2: set confidence

of confidence in its predictions based on the patterns it has learned from the training data. We also have drawn our face landmarks, left-hand and right-hand landmarks, and our pose landmarks with the drawLandmarks function, as shown in Figure 3.3. We used these landmarks to make our detection

```
def draw_landmarks(image, results):
    mp_drawing.draw_landmarks(image, results.face_landmarks, mp_holistic.FACEMESH
    mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_holistic.POSE_CON
    mp_drawing.draw_landmarks(image, results.left_hand_landmarks, mp_holistic.HAN
    mp_drawing.draw_landmarks(image, results.right_hand_landmarks, mp_holistic.HA
```

Figure 3.3: Screenshot of draw landmarks function

3.3.1.2 Data collection for yolov5 model

To train our model yolov5 we had to create our own data , using open cv we collect frames of signs in real time 15 frames for each of the 120 signs .

Figure 3.4: collected frame

Following the unsatisfactory outcome of the initial training, we decided to undergo re-training using alternative and expanded datasets. In pursuit of this objective, we conducted an extensive search in kaggle.[2]and we found the ideal resource to enhance the

[2]https://www.kaggle.com/

training process: WLASL .[3] is the largest video dataset for Word-Level American Sign Language (ASL) recognition, which features 2,000 common different words in ASL. However, we opted to utilize a subset of just 130 words from this comprehensive dataset for the re-training process. This decision aimed to strike a balance between data quantity and manageability while ensuring that the chosen subset encompassed a representative range of words.

3.3.2 Data preprocessing

In this step, the raw data is transformed and prepared for training our models

3.3.2.1 Data preprocessing for lstm model

We extracted landmarks for each sign including pose, face, left hand, and right hand. These landmarks represent specific points or features within the sign gestures. Once the landmarks are extracted we concatenate them to form a unified representation of the sign then we create label map this label map associates each key point, obtained from the concatenated landmarks, with its corresponding label or class. The label indicates the specific sign gesture or meaning associated with that key point. Then we splite them into train and test using **sklearn.model.selection**

```
array([[[ 0.45632309,  0.44088095, -0.57777178, ...,  0.87990999,
          0.74469733, -0.06885431],
        [ 0.48094255,  0.43766466, -0.54148257, ...,  0.11931239,
          0.5651713 , -0.1253465 ],
        [ 0.48879489,  0.43669528, -0.48173547, ...,  0.14751208,
          0.5655551 , -0.11400782],
```

Figure 3.5: landmarks representation

[3] https://www.kaggle.com/datasets/risangbaskoro/wlasl-processed

```
label_map
{'aLL DONE': 0,
 'ask you': 1,
 'baby': 2,
 'bad': 3,
 'book': 4,
 'bored': 5,
 'boy': 6,
 'car': 7,
 'cat': 8,
 'coffee': 9,
 'come': 10,
 'cook': 11,
 'do not want': 12,
 'eat': 13,
 'english': 14,
 'excited': 15,
 'father': 16,
```

Figure 3.6: label map

3.3.2.2 Data preprocessing for yolo v5 model

The "wsal" folder contains the essential file "WLASLv0.3.json," which serves as a repository for the glossary and video instances. In addition, the folder includes a "videos" subdirectory consisting of approximately 12,000 videos, each uniquely identified by its corresponding videoid. In order to enhance organization and facilitate easy access, we have renamed the videos based on their IDs and the corresponding sign language information extracted from the JSON file, ensuring that each video is labeled with its appropriate sign word. Afterward, we utilized OpenCV to extract frames from each video .We selectively choose valid frames from the videos, specifically 100 frames for each sign. These chosen frames we labeled them using labelimg.

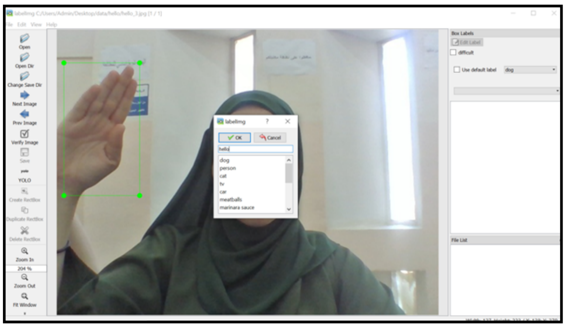


Figure 3.7: Screenshot of LabelImg tool

```
hello_1 - Bloc-notes
Fichier  Edition  Format  Affichage  Aide
60 0.233594 0.493750 0.364063 0.391667
```

Figure 3.8: result of labelImg


```
"gloss": "book",
"instances": [
    {
        "bbox": [
            385,
            37,
            885,
            720
        ],
        "fps": 25,
        "frame_end": -1,
        "frame_start": 1,
        "instance_id": 0,
        "signer_id": 118,
        "source": "aslbrick",
        "split": "train",
        "url": "http://aslbricks.org/New/ASL-Videos/book.mp4",
        "variation_id": 0,
        "video_id": "69241"
    },
```

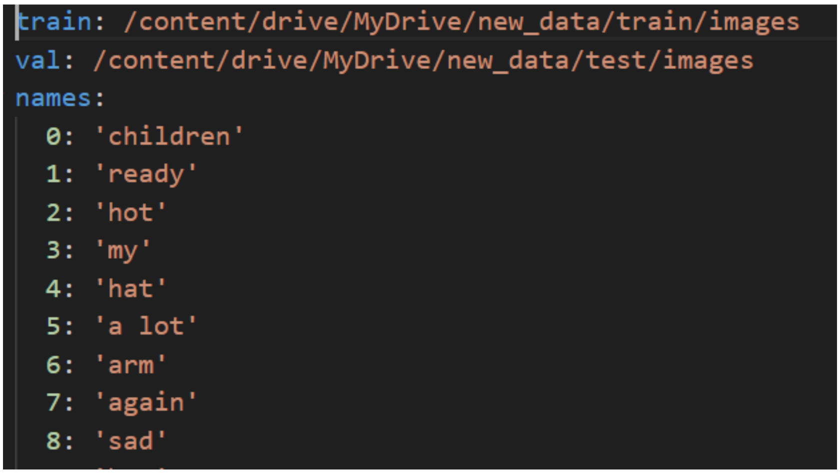

Figure 3.9: Video characteristics in a JSON file

then we create two separate folders: one for training data and another for validation data. These folders will hold the images and their corresponding annotation files then we move a portion of the dataset to the training folder for model training, while another portion is placed in the validation folder for performance evaluation In addition, we create a YAML[4] file that specifies the dataset configuration and training parameters. This YAML file should include information such as the paths to the training and validation data folders, the class labels, and any other relevant settings for training the YOLOv5

[4] https://yaml.org/spec/1.2.2/

model and place it in the main directory then we uploaded this folder onto Google Drive to utilize it for training our model in Google Colab.

Figure 3.10: main directory of training in drive



```yaml
train: /content/drive/MyDrive/new_data/train/images
val: /content/drive/MyDrive/new_data/test/images
names:
  0: 'children'
  1: 'ready'
  2: 'hot'
  3: 'my'
  4: 'hat'
  5: 'a lot'
  6: 'arm'
  7: 'again'
  8: 'sad'
```

Figure 3.11: Screanshot of Yaml file

3.3.3 Training of the models

The training process involves adjusting the model's parameters and optimizing its performance .

3.3.3.1 LSTM neural network building and training

we initialize our sequential model and adjust the parameters and we select optimizer and loss function then start the training prosses.

```python
model = Sequential() #initialize our sequential model
#30,1662
model.add(LSTM(64, return_sequences=True, activation='relu', input_shape=(30,1662))) #our first layer
model.add(LSTM(128, return_sequences=True, activation='relu'))#our Second layer
model.add(LSTM(64, return_sequences=False, activation='relu'))#our third layer
model.add(Dense(64, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(actions.shape[0], activation='softmax'))
```

Figure 3.12: model initialization

```python
#1- Adam
model.compile(optimizer='Adam', loss='categorical_crossentropy', metrics=['categorical_accuracy'])
# we can change the optimazer option

model.fit(X_train, y_train, epochs=2000, callbacks=[tb_callback])
```

Figure 3.13: start trainning process

3.3.3.2 Yolo v5 model training

Before training the YOLOv5 model, the model's settings need to be configured. This process involves selecting the appropriate model architecture (e.g., 's', 'm', 'l', or 'x'), specifying the dataset path and number of classes, setting the input image size . in our case we use this configuration shown in 3.14 :

- **img**: This parameter sets the image size during training to 640x640 pixels.
- **batch**: This parameter sets the batch size to 16, which means that the training process will process 16 images in parallel during each iteration.
- **epoch**:the number of training epochs. An epoch refers to a complete pass through the entire training dataset.the number of training epochs. An epoch refers to a complete pass through the entire training dataset.
- **data**: the path to the YAML file containing the dataset configuration
- **cfg**:the path to the model configuration file which defines the architecture and hyperparameters of the YOLOv5 model
- **weights 'yolov5s.pt'**: the path to the initial weights file. In this case, the model will be initialized with the pre-trained weights from the yolov5s.pt file.

```
%%time
%cd /content/drive/MyDrive/tr/yolov5
!python train.py --img 640 --batch 16 --epochs 50 --data '/content/drive/MyDrive/new_data/data.yaml' --cfg /content/yolov5/models/custom_yolov5x.yaml
```

Figure 3.14: model configuration

After completing this step, the next stage involves training the model. This process requires approximately 8 hours of training time using GPU hardware. During this duration, the model is fed with data to learn and optimize its performance.

3.3.4 Test and Results

In this section we will delve into the crucial aspects of performance measurement and analysis and test our model

3.3.4.1 Performance measures

Model	Recall	Precision
LSTM model	0.79	0.81
YOLOv5 model	0.98	0.98

Table 3.4: performance measures

According to the results of the table indicate that the YOLOv5 model outperforms the LSTM model in terms of recall and precision. This suggests that the YOLOv5 model has better performance in the task of sign detection compared to the LSTM model. However, it is important to consider the specific context of the task for which the models were trained. The LSTM and YOLOv5 models were trained on different datasets with specific objectives and constraints. Therefore, their performances vary depending on the exact nature of the task and the training data. Additionally, the models may struggle with certain types of signs or in challenging lighting conditions. They may also exhibit false positives or false negatives in certain situations for exemple lstm model can detect signs unlike yolo model .

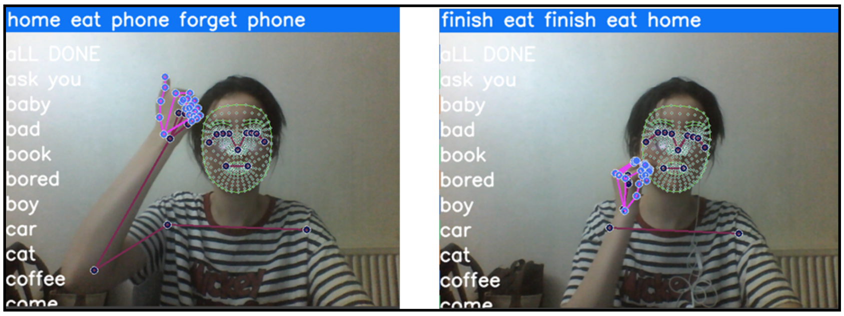


Figure 3.15: sign detection using lstm model

Figure 3.16: sign detection using yolo v5 model

3.4 Signy.io text reformulation

Chapter One mentions that signed language differs from spoken languages like English. Our solution aims to translate the signs to text in the Algerian dialect, our LSTM and YOLOv5 models will give us output phrases in English that illustrate the meaning of each sign as it is shown in Figure 3.17 highlighted in blue where the sentence **"me or mine eat fine eat fine "** is not a structured sentence (no linked words, small vocabulary, and no punctuation and repetitive words "eat fine eat fine"). Consequently, we update to a solution where we reformulate that sentence to get meaningful sentences :

- First, the repetitive words will be removed, as shown in Figure 3.18.

-Second, the sentence will be reformulated to get a coherent one as shown in Figure 3.18.

Figure 3.17: A sentance detected using LSTM model.

```
l = reply.split()
k = []
for i in l:
    # If condition is used to store unique string
    # in another list 'k'
    if (l.count(i) >= 1 and (i not in k)):
        k.append(i)
print(' '.join(k))
```

```
me or mine eat well
```

Figure 3.18: Screenshot of the code responsible of removing repetitive words .

```
messages = [

{"role": "system", "content": "reformulate the sentences"},
]
a=True
while a==True:

    message = input("User : ")
    if message:


        messages.append(
            {"role": "user", "content": message},
        )
        chat = openai.ChatCompletion.create(
            model="gpt-3.5-turbo", messages=messages
        )

    reply = chat.choices[0].message.content
    print(f"ChatGPT: {reply}")
    messages.append({"role": "assistant", "content": reply})
    a=False
    #I do have to work with the replay
```

```
User : me or mine eat well
ChatGPT: Either I eat well or my food is of good quality.
```

Figure 3.19: text reformulation result using GPT API.

3.5 Signy .io machine translation

As mentioned in the previous section, we got a sentence from the reformulation step: "Either I eat well, or my food is of good quality. " We used Google Translate to translate this output to Arabic, shown in figure 3.20. To translate from Arabic to the Algerian dialect, we tried to create a model, but we could not find a dataset that could help us to fulfill our purpose. Thus, first, we attempted to create our dataset, but the phrase's number (738 phrases) was not enough to create one, so as an alternative, we opted for a word-for-word translation where we collected 450 words with their meaning in the Algerian dialect, as shown in Figure 3.21.

```
[14] from typing import Text
     from googletrans import Translator
     translator = Translator()

     a=translator.translate(reply,src=detect_language(reply),dest='ar')
     print(a.text)

     إما أن أتناول الطعام جيدًا أو طعامي ذو نوعية جيدة.
```

Figure 3.20: result of translation with google translate

```
from google.colab import drive
import pandas as pd

# Mount Google Drive
drive.mount('/content/drive')

# Specify the path to your CSV file on Google Drive
csv_path = '/content/drive/MyDrive/PFETAL/amira/Translation_Resources/arabic_to_algerian_1.csv'

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(csv_path)

# Now you can work with the DataFrame as needed
# For example, print the first few rows
print(df.head())

Mounted at /content/drive
  arabic_word algerian_word
0       وقتاش          متى
1          مع           مع
2        صاحب         صديق
3       صحاب        اصدقاء
4    لاكريم       ايس كريم
```

Figure 3.21: word-for-word dataset

```
[ ] def translate_word_by_word(input_text, translation_dict):
        words = input_text.split()
        translated_words = []
        for word in words:
            translated_word = translation_dict.get(word)
            if translated_word:
                translated_words.append(translated_word)
            else:
                translated_words.append(word)
        translated_text = ' '.join(translated_words)
        return translated_text
```

Figure 3.22: word-for-word translation code

3.6 Signy.io Application

To test our solution, we opted for a web application that we developed using nodejs[5], MongoDB[6] , google API [7], and our model that we hosted in hugging face platform .

[5]https://nodejs.org/en

[6]https://www.mongodb.com/

[7]https://console.cloud.google.com/apis

More details about the platform are discussed in the following section.[8].

3.6.1 Signy.io Interface Explanation

The interface is explained in the following table 3.5

Figure	Explanation
3.24	The only page that a visitor can view without using an account. It contains a form that the user fills in to access the platform.
3.25	Redirected to it in case of not having an account. Contain a form to fill out.
3.26	Redirected to it after a successful login. Contain the main functionalities of the platform.
3.27	Give the ability to set the account information.
3.28	Redirected to it after pressing the pencil icon. It gives the ability to set the first name.
3.29	redirected to it after pressing the dataset icon on the home page. Give the ability to contribute to the Algerian dialect Dataset.

Table 3.5: Interface explanation

[8]huggingface.com/

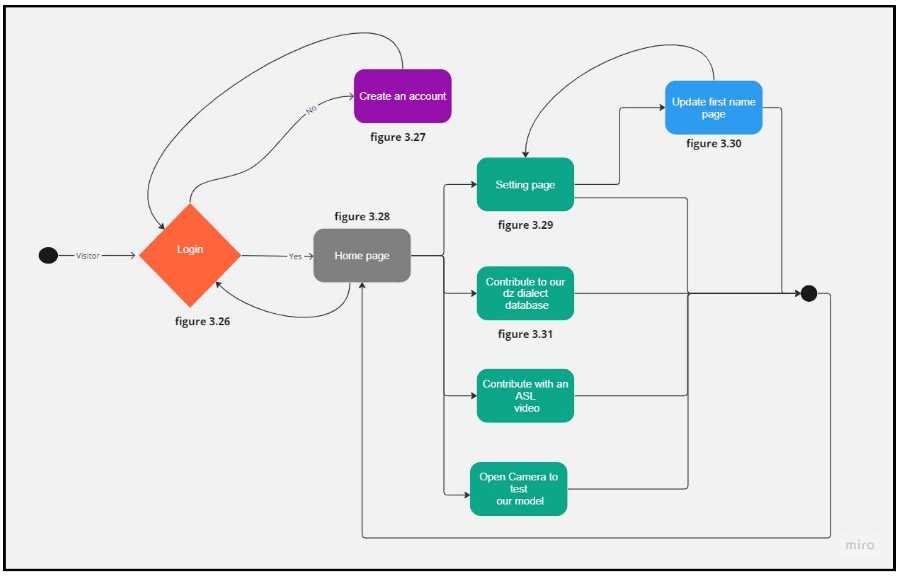


Figure 3.23: Signy.io pages architecture

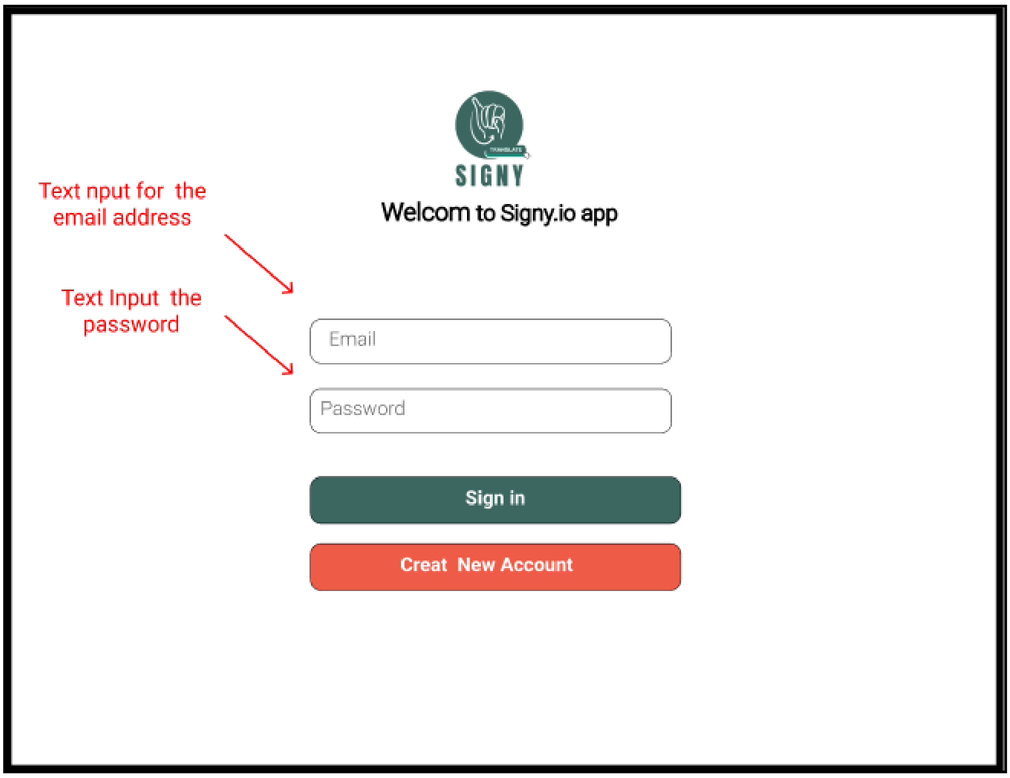


Figure 3.24: Login page

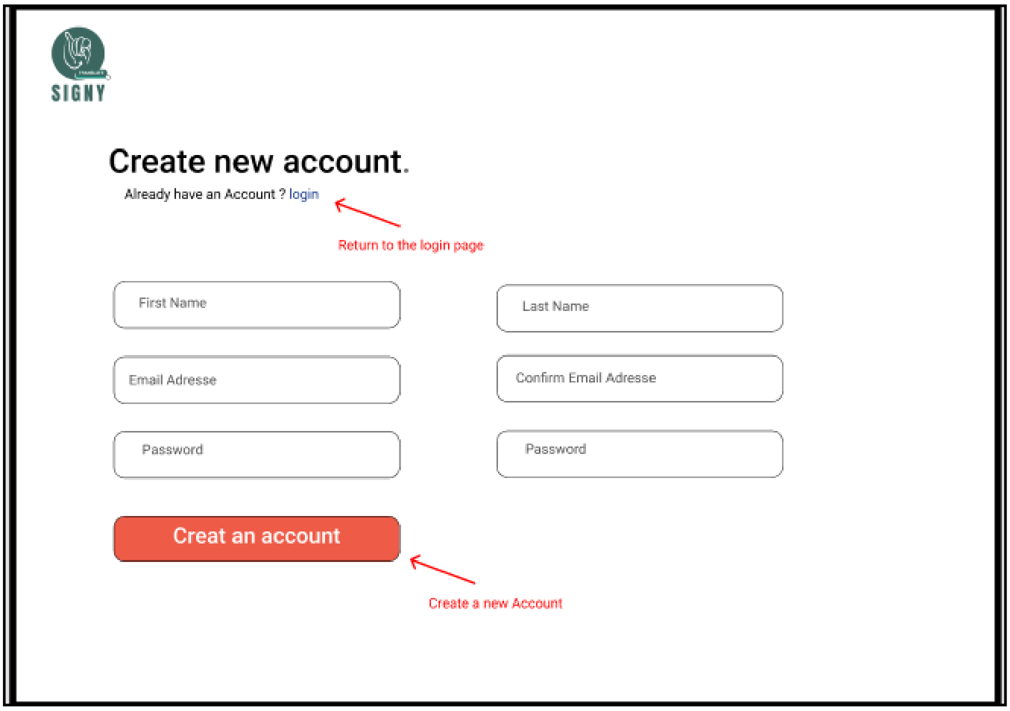


Figure 3.25: Register page

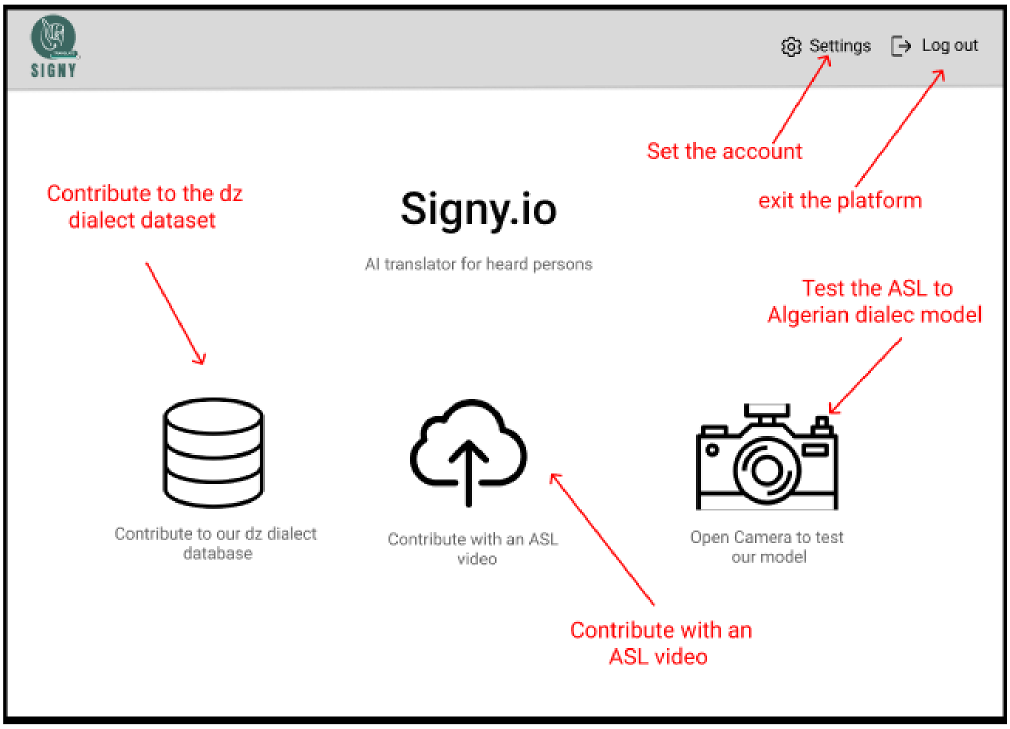


Figure 3.26: Home page

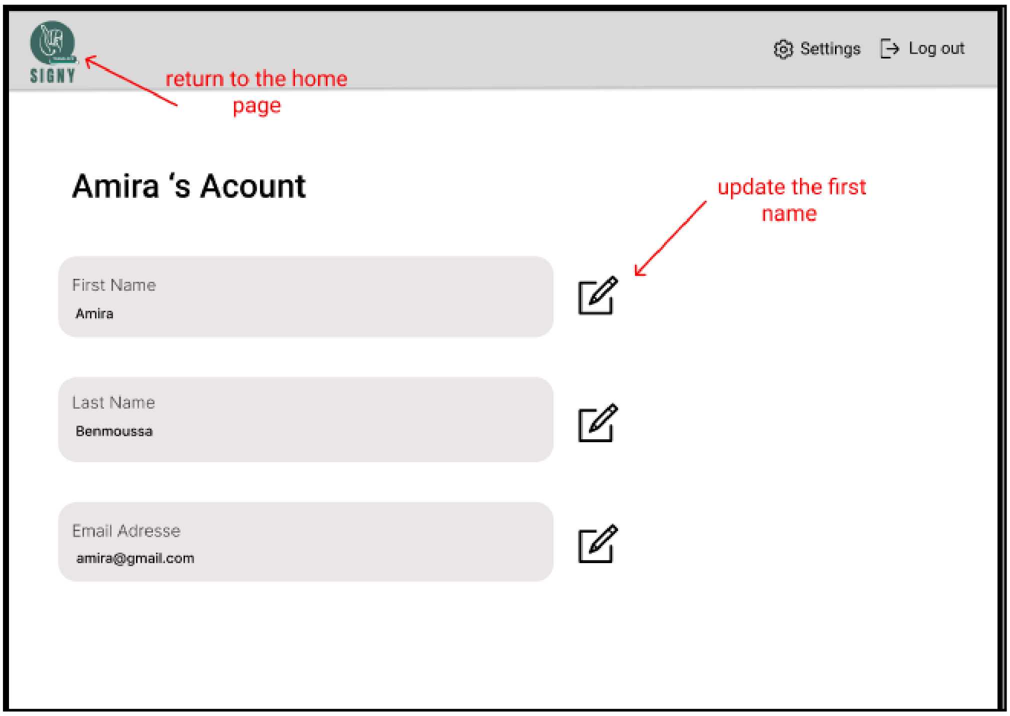


Figure 3.27: setting page

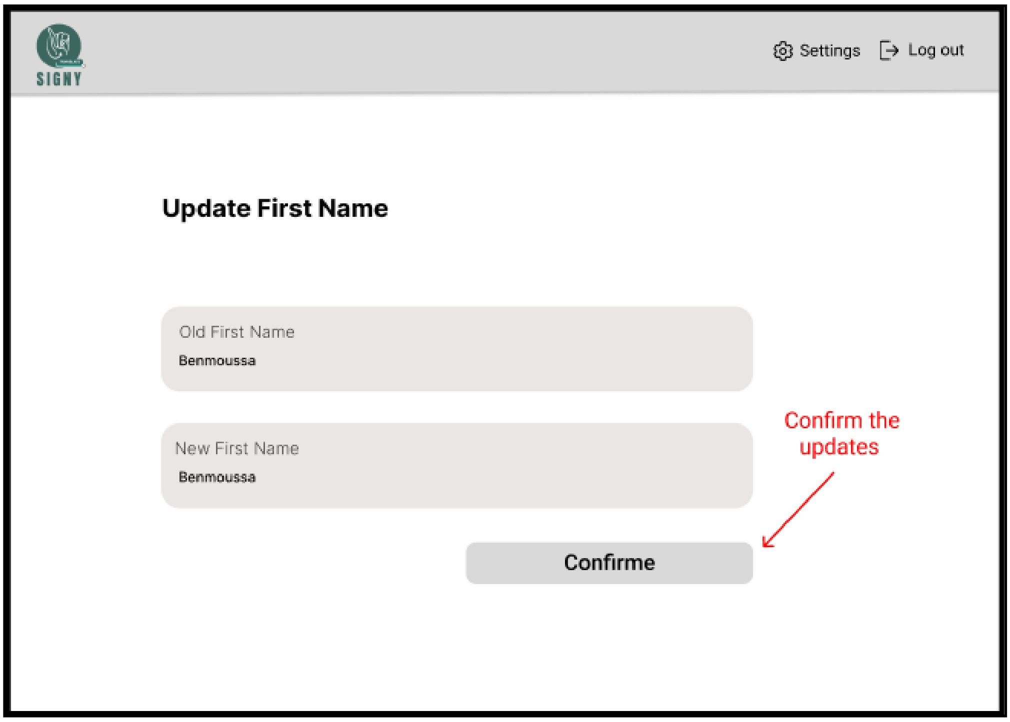


Figure 3.28: first name sitting page

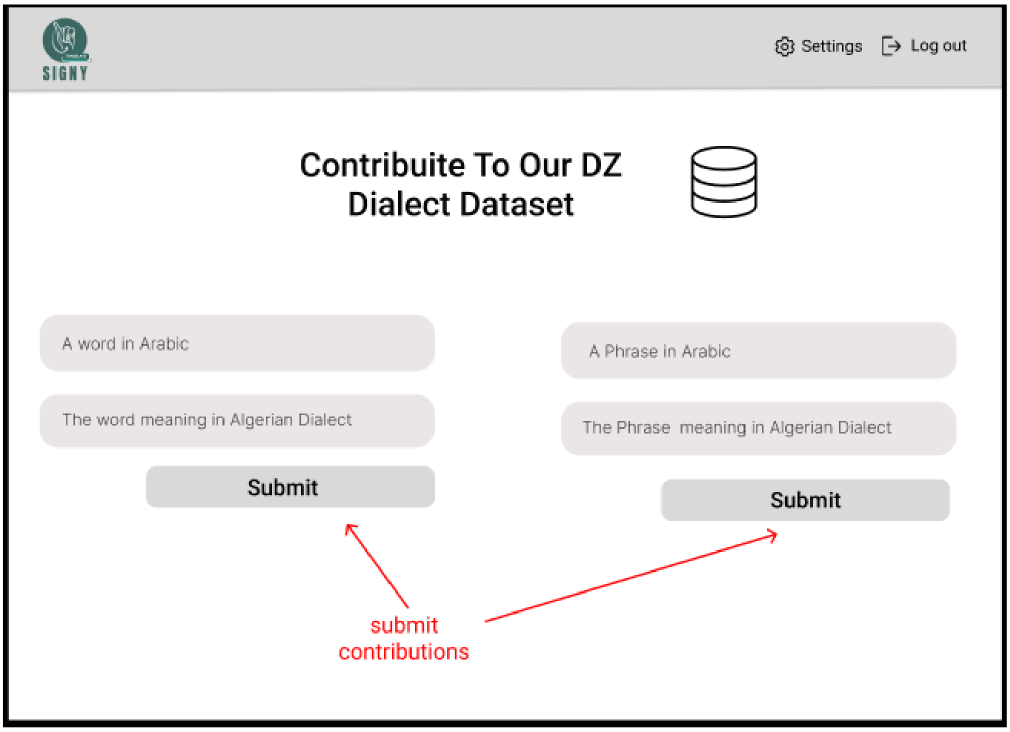


Figure 3.29: Algerian dialect Dataset Contribution page

General Conclusion

This study contributes to the field of sign language translation and recognition by addressing the translation of American Sign Language (ASL) into the Algerian dialect, recognizing the need for localized communication solutions for the deaf community. The development of a sign language recognition model capable of detecting signs in images is a valuable contribution, despite limitations in real-time detection. The utilization of machine translation techniques to translate recognized signs into the Algerian dialect contributes to bridging the communication gap between sign language users and speakers of the Algerian dialect. This approach serves as a foundation for future improvements in dynamic translation methods.

The research project has identified several limitations that should be acknowledged. Firstly, the model of sign language detection lacks real-time capabilities due to limited data availability and hardware constraints. Although it demonstrates accurate recognition on static images, it falls short of real-time interaction, which is crucial for seamless communication. Secondly, the translation phase of the project relies on static machine translation, primarily due to time constraints and inadequate data for training a transformer model. This limitation restricts the translation quality to a word-by-word approach, which may not capture the nuances and idiomatic expressions of the Algerian dialect. It is important to recognize that dynamic translation, considering the context and linguistic variations, would significantly enhance the overall translation accuracy and fluency.

Despite the limitations, there are several avenues for future improvements in this research. To address the real-time sign language detection limitation, acquiring a larger dataset and utilizing more powerful hardware can enhance the model's capabilities. By training the model on video sequences, it can be extended to real-time sign language recognition, enabling more natural and immediate communication. In terms of translation, collecting a more comprehensive dataset specific to the Algerian dialect and training a transformer model would allow for more accurate and context-aware translations. Additionally, exploring the integration of linguistic and cultural factors in the translation process can help capture the richness and subtleties of the Algerian dialect. Furthermore, investigating the development of a hybrid approach that combines the strengths of machine translation APIs with custom models trained on specific sign language and dialect data could offer a more flexible and adaptive translation system. Lastly, the future expansion of the research to include user feedback and evaluation would provide valuable insights into the usability and effectiveness of the developed system, leading to iterative improvements and optimizations.

In conclusion, while acknowledging the limitations imposed by data scarcity, time constraints, and technical challenges, this research makes valuable contributions to sign language translation and recognition. The identified limitations provide a roadmap for future research to overcome these obstacles, leading to more advanced and comprehensive solutions in the field of sign language communication.

Bibliography

[1] World hearing day 2023. https://worldhearingday.org/. Accessed on: June 4, 2023.

[2] britannica. Iraq children.

[3] nationalgeographic. Sign language gestures.

[4] National institute on deafness and other communication disorders. I love you sign.

[5] Mourad Abbas Walid-Khaled Hidouci Kamel Smaïli Salima Harrat, Karima Meftouh. An algerian dialect: Study and resources. *hal.science*, page 14, September 2017.

[6] Koli A. Khatter K. et al Khurana, D. Natural language processing: state of the art, current trends and challenges. *springer.com*, July 2022.

[7] Prabhishek Singh Vijendra Singh Seifedine Kadry Jungeun Kim Sonali Sharma, Manoj Diwakar. Machine translation systems based on classical-statistical-deep-learning approaches. *mdpi.com*, April 2023.

[8] Shu-Hao Chang Cheng-Chia Huang Hao-Che Ho, Kun-Che Chan. Machine translation systems based on classical-statistical-deep-learning approaches. *Journal of Hydroinformatics*, page 18, May 2023.

[9] Angela and Kezhan Shi. How to define a neural network as a mathematical function.

[10] Prabhu. Understanding of convolutional neural network (cnn) — deep learning. [Online].Available:`https://medium.com/@RaghavPrabhu/`

understanding-of-convolutional-neural-network-cnn-deep-learning-
March 2018. Accessed on: June 4, 2023.

[11] Deny. Recurrent neural networks tutorial, part 1 – introduction to rnns, September.

[12] Gesture recognition task guide. [Online].Available:https://developers.google.com/mediapipe. Accessed on: June 18, 2023.

[13] Kangjie Lu Lin Cao Yunfei Liu Renjie Xu, Haifeng Lin. A forest fire detection system based on ensemble learning. *www.mdpi.com*, page 16, February 2021.

[14] Hasim et al Sak. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *research.google*, page 5, 2014.

[15] Quoc V. Le Ilya Sutskever, Oriol Vinyals. Sequence to sequence learning with neural networks. *arxiv.org*, page 3104–3112, August (2014.

[16] Zhifeng Chen Yonghui Wu, Mike Schuster and all. Google's neural machine translation system: Bridging the gap between human and machine translation. *arxiv.org*, October 2016.

[17] United Nations. International day of sign languages.

[18] United Nations. Universal declaration of human rights.

[19] World Federation of Deaf. Who we are.

[20] markoum mina. More than 71,000 people are deaf in algeria. *ennaharonline.com*, page 1, April 2008.

[21] World Health Organization. Deafness and hearing loss. [Online].Available:https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. Accessed on: June 3, 2023.

[22] World Health Organization. World hearing day.

[23] What is sign language?

[24] INÉS ANTÓN DAYAS. How monks helped invent sign language. *https://www.nationalgeographic.com/*, page 1, MAY 2019.

[25] Michelle Jay. History of sign language – deaf history. *www.startasl.com*, page 1, February 2021.

[26] National Geography. Sign language.

[27] Hicham Abdelouafi. Teaching sign language to the deaf children in adrar, algeria: A case study of the hearing impaired children‘s school in adrar, 2018.

[28] Republic of Algeria. Official newspaper of the people's democratic republic of algeria international conventions and agreements - laws and decrees orders, decisions, notices, communications and announcements. *www.un.org*, page 15, May 2002.

[29] Rachid Seghir Kenza Khellas. A realistic dataset for algerian sign language recognition.

[30] National Institutes of Health. American sign language.

[31] Oxford Advanced American Dictionary. dialect.

[32] Mourad Oussalahb Mohamed Berrimia, Abdelouahab Moussaouia and Mohamed Saidia. Arabic dialects identification: North african dialects case study.

[33] Mourad Oussalah Abdelouahab Moussaoui Amine Abdaoui, Mohamed Berrimi. Dziribert: a pre-trained language model for the algerian dialect. *arxiv.org*, page 6, December 2022.

[34] Hadhemi Achour Jihene Younes, Emna Souissi and Ahmed Ferchichi. Language resources for maghrebi arabic dialects' nlp: a survey.

[35] Noam Chomsky. *Aspects of the Theory of Syntax*. The MIT Press, 50 th edition, 1965. Accessed on: June 4, 2023.

[36] Tanvir Ahmed Harris Middi Venkata Sai Rishita, Middi Appala Raju. Machine translation using natural language processing. In *MATEC Web of Conferences*, page 6. 2018 International Joint Conference on Metallurgical and Materials Engineering (JCMME 2018), 2019. Accessed on: June 4, 2023.

[37] Akshit Tayade Arpita Halder. Real-time vernacular sign language recognition using mediapipe and machine learning. *International Journal of Research Publication and Reviews*, page 17, 2021.

[38] Andrey Vakunov Andrei Tkachenka George Sung-Chuo-Ling Chang Matthias Grundmann Fan Zhang, Valentin Bazarevsky. Mediapipe hands: On-device real-time hand tracking. *arxiv.org*, page 18, May 2023.

[39] Jacob Solawetz. What is yolov5? a guide for beginners. [Online].Available:`https://blog.roboflow.com/yolov5-improvements-and-evaluation/`, january 2020.

[40] I-Hau Yeh-Yueh-Hua Wu Ping-Yang Chen Jun-Wei Hsieh Chien-Yao Wang, Hong-Yuan Mark Liao. Cspnet: A new backbone that can enhance learning capability of cnn. *https://arxiv.org/*, page 14, November 2019.

[41] Haifang Qin Jianping Shi-Jiaya Jia Shu Liu, Lu Qi. Path aggregation network for instance segmentation. *https://arxiv.org/*, page 11, September 2018.

[42] Saul Dobilas. Lstm recurrent neural networks — how to teach a network to remember the past.

[43] J. Schmidhuber F. A. Gers and F. Cummins. Learning to forget: Continual prediction with lstm. *Neural Computation*, pages 1735 – 1780, October 2000.

[44] N. N. Schraudolph F. A. Gers and J. Schmidhuber. Learning precise timing with lstm recurrent networks. *Journal of Machine Learning Research*, pages 115–143, August (2002.

[45] Quoc V. Le Ilya Sutskever, Oriol Vinyals. Sequence to sequence learning with neural networks. *arxiv.org*, page 17, 2014.

[46] Vatsal. Word2vec explained.

[47] Adam Roberts Katherine Lee-Sharan Narang-Michael Matena Yanqi Zhou-Wei Li Peter J. Liu Colin Raffel, Noam Shazeer. Exploring the limits of transfer learning with a unified text-to-text transformer. *arxiv.org*, pages 1–67, June 2020.

[48] Adam Roberts Katherine Lee-Sharan Narang-Michael Matena Yanqi Zhou-Wei Li Peter J. Liu Colin Raffel, Noam Shazeer. Attention is all you need. *arxiv.org*, December 2017.

[49] Niki Parmar Jakob Uszkoreit-Llion Jones-Aidan N. Gomez Lukasz Kaiser Illia Polosukhin Ashish Vaswani, Noam Shazeer. Attention is all you need. *https://arxiv.org/*, 2017.

[50] Rewon Child David Luan Dario Amodei Ilya Sutskever Alec Radford, Jeffrey Wu. Language models are unsupervised multitask learners. *https://arxiv.org/*, February 2019.

[51] purefluent. Literal translation.

[52] OpenCV. `https://opencv.org/`.

[53] PyTorch. `https://pytorch.org/`.

[54] TensorFlow. `https://www.tensorflow.org/`.

[55] Google Developers. MediaPipe. `https://developers.google.com/mediapipe`.

[56] OpenAI. `https://openai.com/`.

[57] Heartex Labs. labelImg. `https://github.com/heartexlabs/labelImg`.

[58] Google Translate Community. googletrans. `https://pypi.org/project/googletrans/`.

[59] scikit-learn contributors. scikit-learn. `https://scikit-learn.org/stable/`.

[60] NumPy Community. NumPy. `https://numpy.org/`.

[61] Python Software Foundation. Python Documentation - os module. `https://docs.python.org/3/library/os.html`.

[62] Google. Google Drive. `https://www.google.com/intl/fr/drive/`.