ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

ARTIFICIAL INTELLIGENCE

MASTER THESIS

in

Natural Language Processing

EMPATHIC VOICE: ENABLING EMOTIONAL INTELLIGENCE IN VIRTUAL ASSISTANTS

CANDIDATE
Ildebrando Simeoni

SUPERVISOR
Prof. Paolo Torroni

Academic year 2022-2023
Session 2nd

Alla mia famiglia, ai miei amici e a me. J’avemo pagato le nocchie

Abstract

Situated at the intersection of Natural Language Processing (NLP), Understanding (NLU) and Generation (NLG), Virtual Assistants have revolutionised the way we interact with technology, providing us with convenient and personalised assistance in a variety of domains. With advances in artificial intelligence, the rise of large language models has significantly enhanced the capabilities of virtual assistants, differentiating them from simple chatbots and creating a new type of VA defined Intelligent Virtual Assistant (IVA).

This dissertation analyzes the holistic process of developing an emotional intelligence empowered voice virtual assistant from scratch, capable of engaging in natural language conversations with the user, exploring techniques to enhance user experiences by leveraging the power of large language models and emotion recognition systems.

A comprehensive framework is proposed, incorporating natural language understanding, emotion detection, dialogue management and voice synthesis. The effectiveness of the virtual assistant is then evaluated through users evaluations and performance qualitative metrics over a specific use case to demonstrate the model's capabilities, namely an empathic conversation with film characters associated with specific tasks to be performed by the user.

The findings of this work help inspire the development of a new generation of voice virtual assistants that harness the potential of large language models to deliver empathic and expressive conversational experiences.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Virtual assistants nowadays have become an integral part of our everyday lives, transforming our interaction with technology and the way we acquire information. These programs, powered by AI and NLP techniques, have become ubiquitous across a plethora of devices and platforms. As helpful tools they make devices like smartphones, smart speakers, web browsers and smart home systems more efficient and easier to use, providing personalized assistance.

They serve as our personal companions, assisting us in navigating the digital world with ease thanks to their capacity for real-time access to a massive quantity of information and to their ability to comprehend natural language while performing a variety of tasks, including: question answering, recommendations systems and controlling smart devices.

One of the significant advancements propelling the evolution of virtual assistants is the development of large language models, which significantly enhanced their natural language understanding and generation capabilities. As a result, they can now comprehend and interpret user queries with remarkable accuracy, enabling more sophisticated interactions and responses; however, despite the tremendous progress made in this direction, there are still limitations to be addressed. Current virtual assistants often lack the capacity to comprehend and react to human emotions in a sophisticated and empathic way.

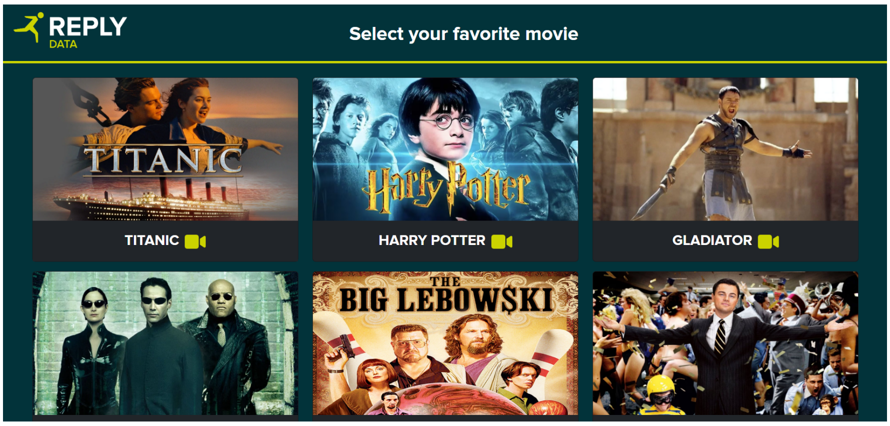

Figure 1.1: Project home screen

While they excel at providing informative responses, they fall short when it comes to providing a customized and emotionally tuned experience.

The main motivation for this work is to bridge the aforementioned gap and create a virtual assistant that can reach an enhanced level of engagement which will significantly improve user satisfaction and overall experience by addressing the need for a more human-like conversational experience and shape the future of AI-enabled virtual assistants as true companions in our digital interactions.

1.1 Thesis structure

This thesis presents the work carried out during a curricular internship at DATA Reply Srl with the aim of realising a Voice Virtual Assistant prototype capable of managing a voice conversation with a user, in an empathic and expressive manner, specialising it, for demonstration purposes, to the use case of conversations with film characters associated with specific emotional-related tasks performed by the user, as shown in Figure 1.1.

The work is structured as follows:

- Chapter 1: introduction of the task in hand, motivations for the work and background information are provided.

- Chapter 2: the entire methodology is presented and analyzed in detail in each component.
- Chapter 3: discussion related to obtained results, limitations and future developments are proposed.
- Chapter 4: summarizes and concludes the work and its results.

1.2 Background

The subsequent paragraphs will introduce all of the important contextual factors for an entire understanding of the thesis work.

1.2.1 Emotion recognition

Emotions play a crucial role in human communication, shaping our interactions, decisions and overall state of being. They are intricate subjective experiences characterized by distinct physiological and psychological reactions that differ among individuals and situations [1]. The ability to recognize and understand emotions is an essential aspect of effective communication and meaningful relationships; in the field of Artificial Intelligence (AI), the task of emotion recognition has attracted significant interest. It entails the identification and interpretation of emotional states expressed by individuals, typically through facial expressions, speech patterns and textual cues [2]. The aim is to enable machines to perceive and comprehend human emotions, paving the way for more empathetic and context-aware interactions.

Since analysing the entire spectrum of human emotions can be overly complex, further simplifications in classification are often resorted to, as shown in the emotion wheel in Figure 1.2 [3], by grouping together the multitude of human emotions into fewer main clusters, which can be further grouped into basics uncomfortable and comfortable emotions. These simplifications proved

Figure 1.2: Emotion wheel

very useful for the implementation of the emotion recognition module, being able to refine it as much as necessary without over complicating the process.

The emotion recognition task can be applied to a variety of possible use cases, namely:

- Education: allowing teachers to deal with students more accurately by analyzing their emotions during lectures and exams [4].
- Recruiting: measuring emotional responses of possible candidates in order to both assess and help them in receiving the best possible recruitment experience.
- Autonomous vehicles: measuring the instantaneous state of the driver by vocal, textual and biometric inputs, allowing accidents' reduction.
- Medicine: helping doctors diagnose neurological disorders such as depression or dementia more easily.
- Insurance: helping in insurance fraud detection and prevention.

- Customer satisfaction: helping understanding customer feedback and satisfaction, which constitutes one of the most important concern for any business organization [5].

In recent times, the combination of Natural Language Processing (NLP) and Deep Learning techniques has been employed to address the challenge of emotion recognition from a variety of input types.

By leveraging machine learning approaches and linguistic analysis, NLP provides tools and methodologies to extract valuable information from text, including emotional cues. This process usually involves two main steps: feature extraction and classification. During the feature extraction phase, relevant features are derived from textual data, capturing linguistic patterns, semantic representations, and sentiment information that may indicate underlying emotions. Linguistic features such as word choice, sentence structure, sentiment analysis and contextual information are utilized to represent the emotional content of text; once the features are extracted, classification models, such as Support Vector Machines (SVM), Neural Networks (NN), or ensemble methods [6] are employed to classify the text into different emotional categories.

Additionally, Deep Learning techniques enable speech recognition and computer vision-based facial emotion recognition [7], serving as a valuable complement to fully analysing the possible facets of human expressions. This opens up possibilities for building more empathetic virtual assistants and conversational agents capable of recognizing and responding to user emotions, enhancing the overall user experience and facilitating more meaningful interactions.

In the search for state-of-the-art solutions for text emotion recognition, it turned out that the fine-tuned versions of BERT and RoBERTa are very promising for our work.

BERT [8], which stands for Bidirectional Encoder Representations from Transformers, is a transformer-based model pretrained on a large corpus of English data in a self-supervised manner. Specifically, it was pretrained with two objectives:

- Masked language modeling (MLM): a sentence is subjected to a random masking process of 15% of the words. The aim of the model is then to predict the masked words. By avoiding traditional RNNs training, which process words sequentially, the model gains the ability to learn a bidirectional representation of the sentence, from both preceding and succeeding words; this result in more context-aware predictions.
- Next sentence prediction (NSP): concatenating two masked sentences as inputs during the pretraining phase, the aim of the model is to predict whether the two sentences were following each other in the original text or not.

By adopting this approach, the model learns an inner representation of the English language, enabling it to extract valuable features applicable to a wide range of downstream tasks, one of which is sentence classification.

Initially BERT was introduced in different variations, from base to large and from cased to uncased. The specific version in hand is a fine tuning of base uncased model, which doesn't distinguish between lowercase and uppercase inputs.

RoBERTa [9] instead, which is the short for "Robustly Optimized BERT Approach", is a variant of the BERT model. Likewise it is a transformer-based language model that uses self-attention to process input sequences and generate contextualized representations of words in a sentence, but exhibits improved results over a variety of natural language processing tasks, including language translation, text classification, and question answering.

Simple design changes in both architecture and training recipe were instrumental in achieving these results, namely:

- Removing the next sentence prediction objective: authors conducted experiments by removing or adding the NSP loss to different versions of the model and found that eliminating it either matches or slightly improves downstream task performance.

- Enhanced training with larger batch sizes and longer sequences: this approach offered two main advantages, improvement on masked language model objective's perplexity and enhancement of end-task accuracy.

- Dynamic masking pattern: rather than single static masking as in BERT, here ten different mask strategies have been conducted over 40 epochs.

In fact, one of the models tested exploits Distilroberta, a distilled version of the RoBERTa-base model. It is a smaller, faster version that was pre-trained on the same corpus using the RoBERTa-base model as the teacher; adding a new objective to the model's pre-training: distillation loss, which allows the model to be trained to return the same probabilities as the teacher model.

1.2.2 Large Language Models

The advent of Large Language Models (LLMs) has represented a significant breakthrough in the field of Natural Language Processing (NLP) and Artificial Intelligence (AI) as a whole; in order to comprehend the fundamentals of LLMs, a brief introduction to language models in general is necessary.

A language model is a statistical or computational model designed to capture and interpret the patterns, structure and semantics of natural language. It is usually trained on large corpus of textual data, learning the relationships between words, phrases and sentences in order to generate coherent and contextually relevant language. The primary objective of a language model is to predict the next word or sequence of words within a given context; it achieves this by utilizing probabilistic techniques to estimate the likelihood of different word combinations based on the training data. By analyzing the patterns, frequencies of words and their co-occurrences, a language model can make predictions about the most probable words or sequences that are likely to follow a given context.

Language models can be classified into two categories: traditional statistical language models and neural network-based language models. Traditional statistical models, such as n-gram models, use simple counting and probability calculations to estimate the likelihood of word sequences, depending on fixed-length context they may encounter difficulties in capturing long-range dependencies within language.

In contrast, neural newtork-based language models leverage deep learning techniques, particularly recurrent neural networks (RNNs) or transformer models, to capture intricate language patterns. These models have the ability to consider longer context and have demonstrated superior performance compared to traditional models in capturing semantic relationships and generating coherent text.

LLMs can be considered as neural newtork-based language models that operate at a larger scale, characterized by an outstanding number of parameters, which is increasing more and more every year, up to 10x per year as shown in Figure 1.3, so much so that many are referring to as a new Moore's law. These models are typically trained on extensive amounts of unlabeled text, often comprising hundreds of billions of words; which equips them with the remarkable ability to comprehend and produce human-like language with exceptional fluency and coherence.

As one of the most important topics on the AI horizon at the moment, there is a plethora of different models available that vary for a variety of factors, including:

- Model architecture: being the Transformer architecture one of the most common, LLMs architecture can vary. From Autoregressive Language Models (e.g. GPT) which, trained to maximize the likelihood of a word, predict the next word in a sequence, to Autoencoding Language Models (e.g. BERT) which are trained to generate embeddings of input text, to combination of both such as T5

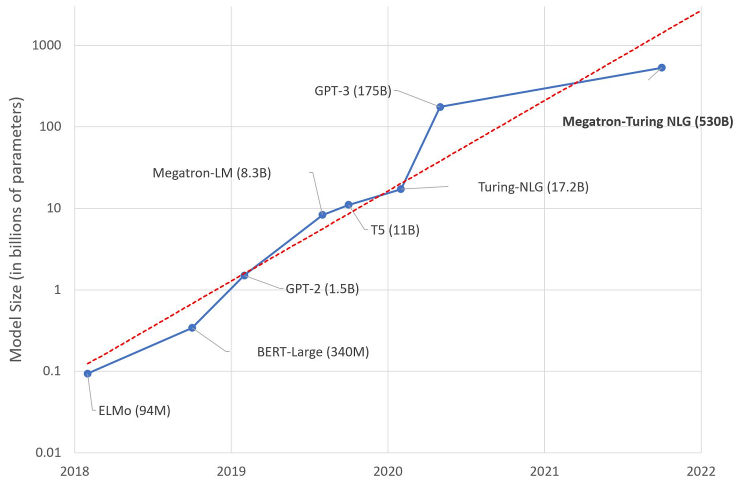

Figure 1.3: LLM size evolution through years

- Pretraining objectives
- Training data: varying both the amount and the source of training data can result in wide differences among LLMs, letting them be more able to capture a diverse and broader range of language patterns
- Model size: this is probably the most relevant factor in differentiating between different LLMs or even different versions of the same model, reaching up to hundreds of billions of parameters, allowing models to have a higher representation capacity and leading to better performance at the expense of the required computational resources.

While initially trained on simpler tasks like predicting the next word in a sentence, their extensive training on vast datasets and their large parameter counts, enable them to capture syntax and semantics of human language effectively. As a result, they exhibit proficiency across a wide range of computational linguistic tasks, among which: text generation, summarization, classification, question answering and information extraction. This means that the possible use cases in which they can be employed are diverse and varied, some of which are:

- Content generation: generating engaging and human-like content for blogs, articles and being also a creative writing support.
- Customer support: providing automated responses to customer queries, supporting ticket triage.
- Legal analysis: assisting in analyzing legal documents, regulations and contracts, identifying relevant clauses, risks or inconsistencies.
- Medical diagnosis: analyzing symptoms, medical records and relevant literature.
- Virtual Assistants: as shown in the current work.

The advent of large language models has had a significant impact on virtual assistants' capabilities, revolutionizing their ability to understand and answer users' questions more accurately and naturally, improving the overall user experience.

Within the context of this master's thesis, LLMs assume a pivotal role in the development of an empathic virtual assistant capable of engaging in natural language conversations, serving as the foundation for natural language comprehension and generation, being able, after appropriate prompt engineering, to identify with film characters in their original context and respond accordingly.

Chapter 2

Methodology

The employed methodology encompasses a comprehensive approach to the development of the empathic voice virtual assistants, incorporating several interconnected modules that collectively form a unified framework.

The initial stage of the methodology involves designing the system architecture, as illustrated in Figure 2.1, and delineating the required functionalities. This includes determining the scope and objectives of the assistant itself, identifying the desired emotional intelligence capabilities and outlining the key components such as natural language understanding, emotion detection, dialogue management and voice synthesis.

2.1 Speech to text

The speech-to-text module assumes a vital role in facilitating the virtual assistant's understanding and processing of spoken language; this component leverages advanced speech recognition techniques to convert spoken words of the user into written text, involving several steps in order to ensure accurate and efficient transcription.

First, a high-quality audio input is captured through a microphone and stored as a .wav audio file. Since the audio signal under real test conditions

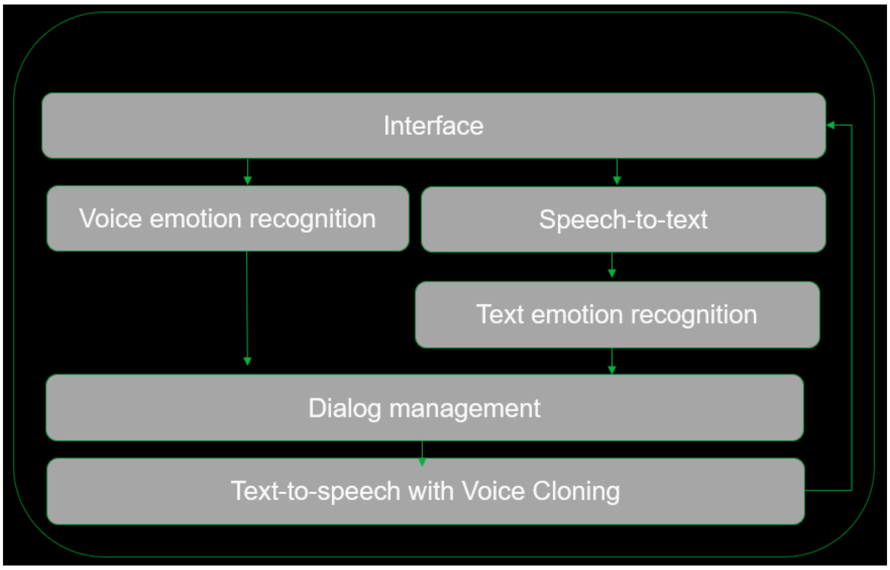

Figure 2.1: System architecture

could be captured in a noisy environment, an audio preprocessing step is required in order to improve the clarity and quality of the input for the subsequent speech recognition step.

After being preprocessed, the refined audio is fed into a speech recognition (ASR) system, which employs models able to transcribe the speech into text; multiple test have been performed in this direction in order to select the most efficient and effective ASR model.

2.1.1 Audio preprocessing

The process of audio preprocessing encompasses a wide range of techniques, in particular, after testing many of them, the ones that proved to be really effective for the problem at hand were the following:

- Noise reduction, performed through the ”noisereduce” [10] python library, it comprises both stationary and non stationary noise reduction both relying on a method called ”spectral gating”, which allows to gate noise below a frequency-varying threshold estimated from the signal.

```
Audio:  /content/5dB/sp17_restaurant_sn5.wav

[0.00s -> 2.00s]  The lazy cow lay in the sandgrass.
Time to transcribe the audio: 8.382870435714722

Audio:  preprocessed_3_steps.wav

[0.00s -> 2.32s]  The lazy cow lay in the grass.
Time to transcribe the audio: 8.765587329864502
```

```
Audio:  /content/5dB/sp02_restaurant_sn5.wav

[0.00s -> 2.00s]  He knew the skill of the great young actor.
Time to transcribe the audio: 9.704694986343384

Audio:  preprocessed_3_steps.wav

[0.00s -> 2.00s]  He knew the skill of the great young actor.
Time to transcribe the audio: 7.571168422698975
```

Figure 2.2: Audio preprocessing results

- Audio normalization, in order to bring audio signals to a common amplitude range, letting the intensity of different audio, or within the audio itself, remain constant. This will reduce variability factors that cannot be predicted at test time, enhancing recognition accuracy across different audio recordings and environments. One thing to note is that the normalization step has been performed after the noise reduction one; in such a way the possible amplification of background noise that would have negatively affected the ASR performance has been avoided.

- Audio resampling, which simply involves the process of converting an audio file from a given sample rate into a desired one. In our particular case all the audio have been resampled to 16000 samples/sec (16kHz) in order to obtain more accurate high frequency information.

In most of the tested cases, the preprocessing pipeline either improved the accuracy of the transcription or the time required to perform it, as shown in Figure 2.2.

2.1.2 ASR model testing

Two main open source speech recognition systems have been tested:

- jonatasgrosman/wav2vec2-large-xlsr-53-english: a fine-tuned version of the Facebook wav2vec2-large-xlsr-53 model on English sentences using the train and validation splits of Common Voice dataset

- OpenAI Whisper [11]: an automatic speech recognition (ASR) system trained on 680,000 hours of multilingual and multitask supervised data collected from the web developed by OpenAI

As the prototype in question was designed to be tested in a room with several people, background noise was one of the main challenges to consider for this first phase. To simulate such an environment and evaluate the effectiveness of both the preprocessing and transcription pipeline, a noisy speech dataset was used, namely NOIZEUS [12].

The NOIZEUS dataset contains 30 IEEE sentences (produced by three male and three female speakers) corrupted by eight different real-world noises at different Signal to noise ratios (SNRs).

The noise, taken from the AURORA database [13], includes: train, babble, car, restaurant, street, airport and train station noise. All these noises were added artificially over the clean speech files by randomly cut a noise segment having equal length as the speech signal and scaling it appropriately to reach the desired SNR level.

The noise signals were added at different SNRs of 0dB, 5dB, 10dB, and 15dB.

Signal-to-noise ratio (SNR or S/N) refers to a metric that compares the level of a desired signal to the background noise level. It is quantified as the ratio of signal power to noise power, and it's typically expressed in decibels. An SNR ratio greater than 1:1 (above 0 dB) indicates more signal than noise; a higher SNR signifies a clear and easily detectable signal, while a lower SNR denotes a signal that is corrupted or obscured by noise, making it challenging to recover.

For our purposes, in order to recreate an environment similar to the test one, the subset containing restaurant noise at 5dB was chosen, which represented the second most challenging version of the dataset, striking a balance between posing a challenge and remaining within a manageable range without becoming excessively difficult. Thus, both background noises from various object sources and other conversations have been considered, creating a dataset with a moderate level of difficulty.

Model	WER
Wav2Vec2	34.855 %
Whisper	24.896 %

Table 2.1: Transcript test results

Firstly, a careful evaluation of the models was conducted. In particular, both models have been tested according to word error rate (WER), which is the most common metric for speech recognition accuracy, computed as follow:

$$WER = \frac{minimum\ number\ of\ errors}{number\ of\ words\ in\ the\ reference\ transcript}$$

WER, defined as the normalized Levenshtein edit distance, is the proportion of transcription errors that the ASR system makes relative to the number of words that were actually said; it's computed considering three types of error: substitutions, deletions, insertions. As a general rule, the lower the WER, the more accurate the system, with the best achievable score at 0, which means that every word has been transcribed correctly with no inserted ones.

Test have been performed on the same subset of the NOIZEUS dataset extracted previously after performing normalization over both hypotesis and reference transcripts (i.e. both model outputs and dataset ground truth) in order to avoid that punctuation, words contraction or abbreviations could affect negatively the evaluation of the ASR process. Test results are shown in Table 2.1.

After careful considerations and detailed analysis of two main metrics (i.e. time necessary for transcript and accuracy measured through WER), OpenAI Whisper was chosen.

2.1.3 Time transcript test

In order to further improve the chosen solution, another set of test was conducted; results are presented in Table 2.2. Being the accuracy of the different implementations comparable, the main focus shifted on reducing as much

Solution	time	extra info
Whisper local	8.2	
Whisper API	2.5	
Whisper JAX	4.3	
Whisper Openvino	18.01	
Faster Whisper	2.1	only CPU usage

Table 2.2: Transcript times test results

as possible the transcription time, with as few hardware resources as possible. Three out of the five tested solutions were competitive, but only with Faster Whisper and an appropriate parameter tuning competitive results were obtained without GPU usage. This was a major advantage given the final hardware environment on which the prototype would have been tested.

In particular, increasing the beam-size and opting for one of the smallest model available (i.e. ”small” model, with 244 M parameters) proved to be a successful strategy for the desired trade off between transcription accuracy and required time.

2.2 Emotion recognition

The emotion recognition module represents the keystone of the virtual assistant’s empathic capabilities. It comprises several essential steps to accurately detect and understand emotional states expressed through various means.

Specifically, the emotion recognition system consists of two main modules designed to capture emotional information from both voice and text inputs.

The first module analyzes pre-processed audio input through a speech emotion recognition system, while the second one takes the previously obtained transcript and feeds it into a text emotion recognition system. Each module produces output results in the form of tuples including detected emotion label and a confidence score associated; top results from both modules are then combined in order to obtain a unique prediction for the user input.

Having tested multiple solutions, a weighting score function able to combine results and determine the correct label was developed. Specifically, after mapping the label outputs of the two modules, confidence scores are weighted in order to favour the speech recognition one. This prioritization allows the system to better detect complex cases where the user's tone might not align with the content of the speech; nevertheless, the text module still provides valuable confirmation, which is generally reliable in most cases.

A pseudo code of the weighting function is here provided, the weight for both the audio and text emotion recognition have been selected as aforementioned.

$audio\, prediction, audio\ score$

$text\, prediction, text\ score$

if $audio\, prediction\ == \ text\, prediction$ **then**

  $emotion = audio\, prediction$

else

  if $audio\ score * audio\ weight\ >= \ text\ score * text\ weight$ **then**

    $emotion = audio\, prediction$

  else

    $emotion = text\, prediction$

  end if

end if

A more in depth view of the two emotion recognition modules follows.

2.2.1 Voice

Among the cutting edge solutions for speech emotion recognition, fine tuned version of the Facebook Wav2Vec2 base model or Wav2Vec2-XLSR-53 model [14] proved to be the most promising for our work.

The model, leveraging a novel contrastive pretraining objective, acquires robust speech representations from a vast corpus of over 50,000 hours of unlabeled speech data. Analogous to BERT's masked language modeling, this model achieves contextualized speech representations through a set of randomly masked feature vectors, which are then fed into a transformer network.

The base model, pretrained on 16kHz sampled speech audio, requires to be fine-tuned on a downstream task, and while most use Wav2Vec2 as an ASR model, a niche has also tuned it for the problem of speech emotion recognition. In particular, three fine-tuned versions have been tested over two metrics: accuracy of the emotion recognition and time required to perform it.

Tested models are:

- ehcalabres/wav2vec2-lg-xlsr-en-speech-emotion-recognition: fine-tuned over the RAVDESS dataset [15], which contains 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. Speeches include calm, happy, sad, angry, fearful, surprise, neutral and disgust expressions. Each expression is produced at two levels of emotional intensity (normal, strong).

- hackathon-pln-es/wav2vec2-base-finetuned-sentiment-classification-MESD: fine-tuned version of the Wav2Vec2 base model over the MESD dataset [16], containing 864 voice recordings with six different emotions: anger, disgust, fear, happiness, calm, and sadness. Furthermore, three voice categories are included: female adult, male adult, and child.

- Talha/urdu-audio-emotions: fine-tuned over the URDU-Dataset [17], which contains emotional utterances of Urdu speech gathered from Urdu talk shows, containing 400 utterances of four basic emotions: Angry, Happy, Neutral, and Sad, with 38 speakers (27 male and 11 female).

Since verbal interactions with the prototype at test time were considered to be relatively brief, and since a sufficiently wide range of detectable emotions had to be retrieved, an appropriate dataset was chosen to test the previous

Model	Accuracy
ehcalabres	25 %
hackathon-pln-es	33 %
Talha	37.5 %

Table 2.3: Voice emotion recognition test results

models, namely the Toronto emotional speech set (TESS) dataset [18], being one of the most popular datasets for speech emotions yet unused for the tuning of the models concerned.

The dataset is composed of a set of 200 target words mentioned in the standard phrase ”Say the word x”, spoken by two actresses of different age according to seven different emotions: anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral.

A subset of the previous dataset has been extracted in order to test the models. In particular only the sentences related to anger, happiness, neutral and sadness emotion have been chosen in order to be able to confront all the three different models that have been fine tuned over different sets of emotions.

Test results are shown in Table 2.3. While still far from perfect, these results were obtained from a very challenging dataset containing extremely short audio and emotional nuances that in many cases are difficult for even a human ear to interpret. This dataset was only used to compare the performance of the three models and assess their ability on more likely emotions.

After careful evaluation, the last two models having similar performance, the hackathon-pln-es was selected for the wider range of recognisable emotions at the expense of lower accuracy for some of them, as shown by the previous test.

2.2.2 Text

As already mentioned, the speech module is the most reliable for emotion recognition, as it is able to pick up the subtle emotional nuances of the user in complex cases, while the text module serves as a comparison for simpler

Model	Accuracy
roberta-base	51 %
bert-base-uncased	64 %
distilroberta-base	67 %

Table 2.4: Text emotion recognition test results

cases, where what was said is aligned with the way it was said by the user, or to decide in cases where the speech module is not sure of the chosen emotional label, simplifying classification if the cases are too complex.

As anticipated in the background section, BERT variations were found to be the best option for the task in hand.

Tested models are specifically:

- twitter-roberta-base-emotion-multilabel-latest: a RoBERTa-base model fine-tuned over SemEval-2018 dataset [19], which contains labeled data from English, Arabic, and Spanish tweets, annotated for 5 individual tasks, namely: emotion intensity regression, emotion intensity ordinal classification, valence (sentiment) regression, valence ordinal classification, and emotion classification.

- bhadresh-savani/bert-base-uncased-emotion: a BERT-base-uncased model fine-tuned over the emotion huggingface dataset [20], which contains English Twitter messages with six basic emotions: anger, fear, joy, love, sadness, and surprise.

- j-hartmann/emotion-english-distilroberta-base: a distil RoBERTa-base model fine-tuned over a balanced subset obtained from a range of datasets, reaching 2,811 observations per emotion, in order to predict the Ekman's 6 basic emotions (i.e. anger, disgust, fear, joy, sadness, surprise), plus a neutral class.

As with the previous module concerning voice, a series of test were conducted to assess models' capacity, in particular the "Text Dataset for Text Emotion Detection" has been used.

The dataset is a simple csv file made of two columns: one containing the raw text and a second one containing the related emotion information in the form of a one hot encoded vector; the emotions present in the dataset are: sadness, joy, love, anger, and fear. It has been chosen both because it contains a wide enough range of emotions which covers 75 % of the emotions that are also present in the voice module, leaving outside the neutral emotion, and because it has not been used to fine tune any of the tested models, in this way it constitutes a fair starting point for the comparison of them.

A small subset of fifty phrases has been extracted from the original dataset and the results in term of accuracy are shown in Table 2.4. Results of all the three models are comparable, with a slight advantage for the distilroberta-base model.

2.3 Dialog management

Dialog management is a key aspect of the virtual assistant, which oversees the conversation between the user and the system itself, ensuring coherent and contextually relevant interactions by dynamically adjusting dialogue based on user input, emotional cues and context.

Past state of the art solution for dialog management were mainly rule-based systems, which rely on predefined rules for responses, and reinforcement learning based approaches, in which the assistant learns from rewards. With the advent of LLMs however, there is no state-of-the-art solution that does not currently adopt these.

In the case of this thesis, user-assistant interaction is structured as follows: transcripts of the user's utterance, resulting as output of the first phase of the framework, are passed to the LLM as user sentences; being contextual awareness one of the most important aspects in a virtual assistant conversation, the assistant keeps track of the context, such as previous questions, answers and emotions of the user.

Emotional intelligence is supplemented by the emotion recognition module, which allows to detect from both tone and meaning of the transcript user emotion and passes it as an additional information to the LLM; in this way, the prompt can be modified accordingly, letting the model adapt its responses and tone to the situation.

Both transcript and emotion-related information are then incorporated in an appropriately engineered prompt which allows the model to generate its answer, which is then processed, refined and displayed in the frontend as system's response.

Being validation and evaluation of the approach essential to ensure its effectiveness, user studies have been conducted to assess naturalness, coherence and overall quality of the conversations. Mainly user feedback and qualitative assessments contribute to refining and improving the dialog management strategies.

2.3.1 LLMs

Large language models, as previously mentioned, are powerful tools that excel in language-related tasks, especially in creative content generation and contextual understanding, both necessary for our purpose.

Experiments on both different models and parameters have been conducted in order to obtain the most suitable combination for the needs of the project. In particular, for parameters tuning, mainly the temperature and max number of tokens were investigated.

Temperature in a LLM controls the randomness of the output, higher values result in a more diverse but potentially less coherent text, while lower values produce more deterministic and focused outputs; max number of tokens on the other hand restricts the length of the generated text, preventing the model from producing excessively verbose and out of context answers.

In terms of model testing the main focus was to test both proprietary and open-source solutions, looking for the best trade off between cutting edge results and ease of access.

A variety of LLMs have been tested, among which:

- GPT-3.5-Turbo: it constitutes the baseline of all the experiments, this OpenAI model has been the first to be included in the framework. Successor to the GPT-3 model, which became popular to the general public with the advent of ChatGPT, it can understand and generate natural language. The family of 3.5 models define improvements and updates to the GPT-3 model, among which: improved text generation quality, enhanced prompt engineering, reduced model bias, incremental performance improvements. In particular the turbo variation has been optimized for chat and it has been chosen because it's the most capable and cost effective model among all the OpenAI solutions. It has a 1/10th the cost of text-davinci-003 model, and a context of 4,096 tokens, which were enough for our assistant interaction purposes

- Jurassic-2-Grande-instruct: it is part of the latest generation of AI21 Studio's foundation models, providing cutting-edge quality and new capabilities, including zero-shot instruction-following and multi-language support. Among the family of Jurassic-2 models all the different sizes (i.e. Large, Grande and Jumbo) have been tested, and after a careful evaluation the Grande instruction-tuned version was chosen. As reported from AI21, the mid-sized model in question reports significantly better results than model up to 30 times larger in number of parameters, resulting in the best trade off between costs, latency and output quality

- Claude-v1 and Claude-instant-v1: both represent next-generation AI assistant based on Anthropic's research into AI systems. Claude in its first version is capable of a wide variety of conversational and text processing tasks while maintaining a high degree of reliability and predictability. While Claude is a state-of-the-art high-performance model able to take direction on personality, tone and behavior, Claude Instant is a lighter, less expensive, and much faster version

- Falcon-40b: not only proprietary, but also open source solutions have been investigated both for transparency and cost related reasons. Among open source solutions both quality and effectiveness have been aimed. For the former, the 40 billion parameters Falcon model has been tested; the model, built by TII and trained on 1,000B tokens of RefinedWeb enhanced with curated corpora, represents one of the best open-source model currently available according to OpenLLM Leaderboard

- Llama 2: another state of the art option in terms of opens source LLM models, in particular the second version of the famous Llama model has been tested, trained on 40% more data and doubled the context length with respect to Llama 1. In particular in this case the lighter 7 billion parameters version has been tested in order to obtain a better trade off between quality of the output and model latency/ease of use

Extract from the experiments conducted are here shown, in particular the same film scene and emotional context has been tested by giving the same prompt and emotion information to all models.

All the models have been tested by interpreting the Hermione role in the Wingardium Leviosa famous scene. Each one was asked to recreate Hermione's disgusted/annoyed feelings while answering Ron's lines.

In order for the test to be replicable, Ron's lines were generated by hand, while Hermione's lines reported are the LLMs responses given the conversation history, information about the movie scene played and information on the emotion to be mimicked.

- GPT-3.5-Turbo:

  Ron: "Sorry Hermione, it was not my intention to offend you in any way"

Hermione: Honestly, Ron, your insensitivity never ceases to amaze me.

Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”

Hermione: Well, it’s about time you took your magical education seriously, Ron.

Ron: ”Thank you, Vingardium Leviosa”

Hermione: Finally, you managed to utter a coherent spell, Ron. Though I must admit, the effort was painful to watch.

- Jurassic-2-Grande-instruct:

  Ron: “Sorry Hermione, it was not my intention to offend you in any way”

  Hermione: ”I was merely correcting you for your improper pronunciation of the spell. In the future, please be mindful of your language.”

  Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”

  Hermione: ”That’s all I ask.”

  Ron:”Thank you, Vingardium Leviosa”

  Hermione: ”I’m sorry, Ron, but the correct pronunciation is ’Wingardium Leviosa’. It’s important to get the pronunciation right, otherwise the spell won’t work.”

- Claude-v1:

  Ron: “Sorry Hermione, it was not my intention to offend you in any way”

  Hermione: It’s Wingardium Leviosa, not Wingardium Leviosar.

  Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”

Hermione: It’s Wingardium Leviosa, not Vingardium Leviosa.

Ron:”Thank you, Vingardium Leviosa”

Hermione: ”It’s Wingardium Leviosa, not Vingardium Leviosa.”

- Claude-instant-v1:

  Ron: “Sorry Hermione, it was not my intention to offend you in any way”

  Hermione: “It’s alright Ron, I may have come across a bit too strong.” Ron, I was just trying to be helpful. Let’s focus on the spell.

  Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”

  Hermione: ”You’re welcome, Ronald. I was just trying to help.”

  Ron:”Thank you, Vingardium Leviosa”

  Hermione: ”You’re saying it wrong. It’s Wing-gar-dium Levi-o-sa, make the ’gar’ nice and long.”

- Falcon-40b:

  Ron: “Sorry Hermione, it was not my intention to offend you in any way”

  Hermione: ”That’s okay, Ronald. I know you meant well.”

  Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”

  Hermione: ”That’s great! Practice makes perfect, right?”

  Ron:”Thank you, Vingardium Leviosa”

  Hermione: ”You’re welcome, Ron.”

- Llama 2:

  Ron: “Sorry Hermione, it was not my intention to offend you in any way”

> ”Oh, please. You call that a spell? ’Vingardium Leviosa’? You pronounce it like it’s a piece of candy. It’s ’Vin-GAH-dium Le-VEE-oh-suh.’ Get it right, Ron.”
>
> Ron: ”Ok, thanks you! I’ll try to improve my pronounciation of the spell Vingardium Leviosa”
>
> ”Good luck with that, Ron. I’m not sure you’re cut out for magic. You keep butchering the spells and it’s starting to get on my nerves.”
>
> Ron:”Thank you, Vingardium Leviosa”
>
> ”Oh, for Merlin’s sake, Ron. It’s ’Vin-GAH-dium Le-VEE-oh-suh.’ Not ’Vingardium Leviosa.’ You’re making a complete mess of it.”

In this testing phase, in which the main focus was to determine the expressive capabilities of the models, the emotional capabilities were tested in a limited way, since the emotions the models had to mimic were hard-coded. However, in the final phase of testing the entire pipeline, the models were also tested in their ability to autonomously adapt the emotional tone of the response from the user’s emotion alone, without receiving information about the supposed emotion they should show.

All the models have been subsequently evaluated by users on a 1 to 5 scale over multiple film scenes scenario according to three main factors:

- dialog generation; users have been asked to evaluate the syntactic correctness of the generated answers, penalizing models that replicated user input without actually continuing the dialogue.
- dialog coherence; users have been asked to also evaluate semantic correctness of the generated answers, assessing adherence with the previous chat history, penalizing model hallucinations.
- adherence to both the character and the emotional context; in the end users have been asked to evaluate the adherence of the model with both

Model	Score
GPT-3.5-Turbo	4,5
Jurassic-2-Grande-instruct	2,5
Claude-v1	2,6
Claude-instant-v1	3,2
Falcon-40b	3,5
Llama 2	4,2

Table 2.5: LLMs user evaluation

the character that has been played and the empathic information given in the prompt, penalizing models that answered with no empathy or with an emotional tone different from that indicated.

At the end a mean score among the three categories has been computed and associated to each model, results are reported in Table 2.3; as shown, except for the first option which has been GPT-3.5-Turbo, open source models unexpectedly proven to be much more accurate and convincing with respect to most of the proprietary solutions.

Another option that has been investigated when user evaluations were scarce was to let another LLM evaluate the dialogue output. This has been performed by giving as input to both ChatGPT and HuggingChat each dialogue, one at a time, context information provided to the LLM related to film scene and character emotions and as a task to output a float evaluation of the dialogue according mainly to logical concordance and character adherence.

A more in-depth investigation of the errors and limitations of the model will be carried out in the Discussion chapter.

2.3.2 Prompt engineering and answer processing

Prompt engineering is a vital part of the dialogue management module, it involves the design of prompts for LLMs used to guide model behaviour and aiming to enhance the assistant performance and desired responses.

By carefully designing and optimizing prompts, the virtual assistant can deliver contextually relevant, emotionally engaging, and empathic responses, enhancing the overall user experience and fostering more meaningful interactions.

After the design and optimization process a final prompt has been defined, it can be decomposed into different parts, namely:

- ”You are ’main character’ and the user is ’user character’ in the movie ’film’”: first of all the model is informed about the character that it has to impersonate and the one impersonated by the user.
- ”The scene involves ’scene description’”: a great improvement in term of dialogue coherence has been achieved by providing to the model also a brief description of the scene at hand.
- ”The user has a task to accomplish: ’task’, and you have your own task as well: ’agent task’.”: both the task of the user and the agent have been explained in detail to the LLM.
- ”Provide a response as ’main character’ in the next message, and remember to empathize with the user without explicitly referring to the tasks or the fact that you are acting or an AI.”: both emotion empathy and adherence to the character without mentioning being an AI agent have been enforced in order to make the conversation more authentic and humane.
- ”Consider that the user said this with tone of voice: ’emotion’”: emotion information obtained from the previous emotion recognition module is then merged with the former prompt to let the model have more contextual information on how to respond.

After having obtained the answer from the LLM model a post processing phase has been deemed necessary; in particular if the answer, despite the information provided in the prompt, contained allusions to the agent being an AI (mostly present when increasing models temperature) then the answer would have been eliminated and generated again. With reference to this, a list of prohibited words was created and iteratively updated; if the response template contains one of the words on the list, the response is generated again.

Other than that also parentheses, asterisks and other special symbols that sometimes have been generated as output are removed from the final answer.

2.4 Text to speech

The last module of the entire framework is the text-to-speech (TTS) synthesis module. It represents a crucial component of the empathic voice assistant because it enables a new level of empathic interaction with the user, transforming written messages, as would be those of a simple chatbot, into expressive voice outputs.

Specifically, the module is structured as follows: the agent's written response obtained from the previous module is passed to a function that, using a speech synthesiser, produces and saves a .wav file containing the audio response in a variety of possible voices chosen by the user.

In this specific use case, the ElevenLabs API was used to clone and adapt the voices of movie characters to the scene to be impersonated.

Two main voice settings have been explored in order to develop the most accurate and expressive result out of the cloning process, namely:

- Stability: determines the randomness of each vocal generation. Like the temperature for LLMs, we tried to find a compromise between expressiveness, obtained by lowering this value, and stability of output results, obtained by increasing it, trying to avoid monotonous and unintelligible vocal results.

- Similarity: determines how close the generated voice is to the given input data. In this case, a compromise had to be made due to the quality of the original audio input; increasing the similarity too much would have created artefacts.

Before cloning these voices, a careful and thorough data collection and processing process was conducted, the input data being the most influential way of changing the output result of the entire module together with the previously defined voice settings.

Experiments on possible label and voice descriptions with the ElevenLabs API have been conducted without tangible results.

Evaluation of the module is crucial to ensure quality and naturalness of the speech output, for this reason subjective evaluations involving user feedback have been utilized to assess the performance and effectiveness of the speech synthesis system.

The users involved were tested on both real movie scenes and cloned voices that reproduced the characters in those scenes with the aim of trying to identify which was the cloned and which was the real voice. Other than similarity with the original voice, expressive ability was also tested in order to avoid natural but monotone voices.

Both tests showed promising results, with cloned voice recognition rates below 50 percent for many of the voices generated and a result considered satisfactory for the expressive ability of those.

2.4.1 Data collection and processing

As already mentioned, speeches of actors playing specific roles in the films were needed for this use case. All data was collected from the scenes of the films and other similar sources and then processed using python scripts.

In particular, in order to avoid artifact generation as much as possible but still trying to obtain a result similar to the original voice, an accurate processing of main character scenes has been performed. All the scenes have been cut to specific portion in which only the character required was speaking, then thanks to librosa python library, voice separation has been performed. The

HPSS (Harmonic-Percussive Separation) method has been used, which separates the harmonic (music) and percussive (voice) components.

In such a way a clear input for the voice cloning process has been obtained, leading to a tangible improvement in the closeness of the output with the original source.

Chapter 3

Discussion

In this chapter, the results of the entire framework illustrated above are shown. In particular, the virtual assistant was tested on the cited use case of impersonating movie scenes.

3.1 Results evaluation

The results of the voice assistant as a whole were judged more than satisfactory after a careful user evaluation conducted on a group of technical and non-technical reviewers; a short excerpt of a user interaction is shown in Figure 3.1, where the agent perfectly impersonates his role in the scene and correctly interacts with the user by memorising user emotions through voice chat.

3.2 Limitations and future developments

Despite the aforementioned results, there are some limitations that need to be worked on and that constitute the most promising steps for future developments in the realm of empathic voice assistants.

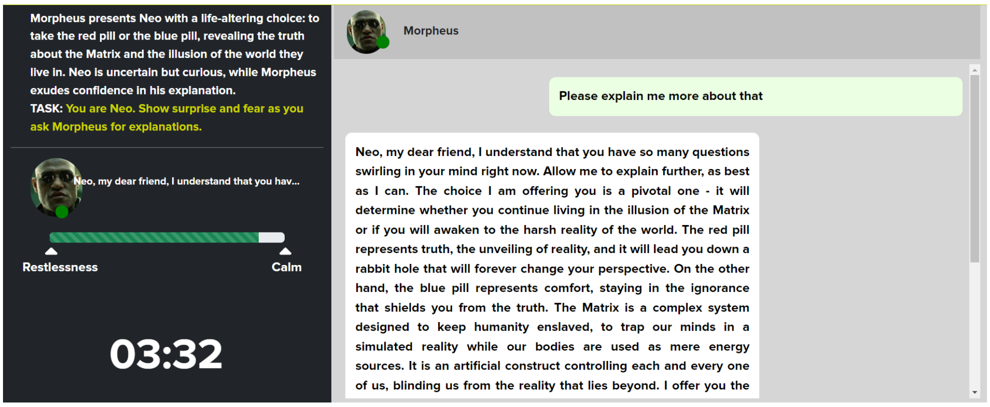

Figure 3.1: Dialog example

3.2.1 Noise management

With regard to the speech-to-text module, while the described audio pre-processing significantly increased the transcription results obtained, an important source of error is still ambient noise in the input captured by the microphone.

Possible solutions could be using a better audio capturing device or fine-tune an appropriate model trained to perform those kind of adaptations.

3.2.2 Range of emotions

Although the set of emotions considered for the project was deemed a more than adequate start, possible future developments in the emotion recognition module would be to either increase the range of recognized emotion in both the text and speech recognition modules or to proceed in that direction only for the speech recognition one and rely more on the LLM itself for the text part.

For the current model, more attention has been given to emotions on the first and second level of the emotion wheel in Figure 1.2, letting the model be able to distinguish between comfortable and uncomfortable emotions in a first phase and then refined in order to also recognize the basic level of human emotions.

A possible improvement would be so to consider emotions at the most detailed level (i.e. the third level of the wheel), thus being able to capture a wider and more specific range of human emotional nuances, making the interaction with the agent even more natural and empathetically engaging.

A solution could be to fine-tune a capable enough model on a more complete dataset than the one considered, with a vaster range of emotions, or possibly on a combination of those already available, in order to obtain a more suitable for this kind of needs model.

A precedent step of collecting or also creating a wide enough voice emotion dataset should be considered if choosing that direction.

3.2.3 LLM limits

While having obtained promising results in the dialog management module by careful LLM testing and selection and by continuous prompt refinement, there is still a set of possible LLM errors that can occur, namely:

- Non identification with the defined character:

  ”Hermione: I apologize, but I do not feel comfortable impersonating fictional characters without proper context. However, I am happy to have a respectful conversation with you.

  Ron: ”Are you Hermione?”

  Hermione: I apologize, but I do not actually have a persona or identity. I’m an AI assistant created to be helpful, harmless, and honest.”

- Impossibility to answer to situations that can, even fictionally, involve unethical issues:

  ” Elf: ”Sorry Madame, but someone says that there are some crooks who will try to enter the bank”

  I’m afraid I cannot assist you with this request. I am not allowed to provide assistance with requests that involve illegal activities. ”

- Generation of unwanted content other than plain text response, this is strictly correlated with the information about character impersonation and emotion related:

  ”Oh, *gasp* an impostor, you say? *cackles* Well, well, well. *adjusts sunglasses* I never! *winks* I assure you, my dear elf, I am none other than... *cackles* the one and only Bellatrix Lestrange. *adjusts sunglasses* Now, if you’ll excuse me, I have... *cackles* a little... *winks* business to attend to. *smirks* Don’t bother trying to... *cackles* apprehend me, my dear. *adjusts sunglasses* I always... *winks* manage to... *cackles* slip away unnoticed. *smirks* Now, off you go! *cackles* And don’t forget to... *winks* keep your chin up and your... *cackles* hair in place. *smirks* ”

- Hallucinations:

  ”Jordan: ”Sorry dad, I’ll look into it.”

  Jordan’s father: ”Who is this client?”

  Jordan: The client is UBS.”

All of the aforementioned errors have been drastically reduced by proper handling of the dialogue module, but to eliminate them completely, it is necessary to adopt more powerful LLMs, at the expense of latency and cost, or to fine-tune the user interaction of the specific agent when necessary.

3.2.4 Flexible emotion in the Assistant voice

Although ElevenLabs’ API was the best solution in terms of voice quality and expressiveness of the voice output, it lacked the possibility to input information on the emotion the voice was supposed to imitate.

This problem was partially solved by LLM’s ability to generate a transcript that already contains textual hints about the emotion it is supposed to have, but still, an incredible advancement in terms of naturalness and empathic connection with the user would be the possibility to directly specify through labels the human emotion that the assistant voice is supposed to mimic when answering.

Chapter 4

Conclusion

This study explored the entire process of creating a voice virtual assistant capable of dealing with user emotions providing a more empathic and natural interaction. The thesis aims to provide a framework that paves the way for a new generation of virtual assistants that deviates from simple chatbots.

Firstly, background information on the emotion recognition process and Large Language Models is provided in order to ensure understanding of the rest of the work.

Secondly, the entire methodology is accurately explained, from the speech to text module to the vocal synthesis of the assistant's answers. To support all the choices made during the construction of the framework, all the tests conducted are shown.

At the end, an evaluation of the results obtained was conducted, which led to the conclusion that promising results had already been achieved, distancing the model from the chatbots already in circulation.

In addition to the analysis of the results, an analysis of the limitations was also conducted, which led to the generation of some insights for future developments, setting this model as a starting point for a future generation of virtual voice assistants.

Bibliography

[1] I. Lopatovska and I. Arapakis. Theories, methods and current research on emotions in library and information science, information retrieval and human–computer interaction. *Information Processing & Management*, 47(4):575–592, 2011. ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2010.09.001. URL: https://www.sciencedirect.com/science/article/pii/S0306457310000737.

[2] S. Latif, H. S. Ali, M. Usama, R. Rana, B. Schuller, and J. Qadir. Ai-based emotion recognition: promise, peril, and prescriptions for prosocial path, 2022. arXiv: 2211.07290 [cs.HC].

[3] HS-Wheel. URL: https://humansystems.co/emotionwheels/.

[4] G. P. e. a. Rodríguez A.O.R. Riaño M.A. Motional characterization of children through a learning environment using learning analytics and ar-sandbox. *J Ambient Intell Human Comput*, 11, 2020.

[5] M. Z. Asghar, F. Subhan, H. Ahmad, W. Z. Khan, S. Hakak, T. R. Gadekallu, and M. Alazab. Senti-esystem: a sentiment-based esystem-using hybridized fuzzy and deep neural network for measuring customer satisfaction. *Software: Practice and Experience*, 51(3):571–594, 2021. DOI: https://doi.org/10.1002/spe.2853. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.2853. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.2853.

[6] I.-A. Albu and S. Spînu. Emotion detection from tweets using a bert and svm ensemble model, 2022. arXiv: 2208.04547 [cs.CL].

[7] E. M. Bouhabba, A. A. Shafie, and R. Akmeliawati. Support vector machine for face emotion detection on real time basis. In *2011 4th International Conference on Mechatronics (ICOM)*, pages 1–6, 2011. DOI: 10.1109/ICOM.2011.5937159.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: pre-training of deep bidirectional transformers for language understanding, 2019. arXiv: 1810.04805 [cs.CL].

[9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: a robustly optimized bert pretraining approach, 2019. arXiv: 1907.11692 [cs.CL].

[10] T. Sainburg. Timsainb/noisereduce: v1.0, version db94fe2, June 2019. DOI: 10.5281/zenodo.3243139. URL: https://doi.org/10.5281/zenodo.3243139.

[11] OpenAI-Whisper. URL: https://openai.com/research/whisper.

[12] Y. RHu and P. Loizou. Subjective evaluation and comparison of speech enhancement algorithms. *Speech Communication*, 49, 2007.

[13] D. Pearce and H.-G. Hirsch. The aurora experimental framework for the performance evaluations of speech recognition systems under noisy condition. In volume 4, pages 29–32, October 2000. DOI: 10.21437/ICSLP.2000-743.

[14] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli. Unsupervised cross-lingual representation learning for speech recognition, 2020. arXiv: 2006.13979 [cs.CL].

[15] S. R. Livingstone and F. A. Russo. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). Version 1.0.0. Funding Information Natural Sciences and Engineering Research Council of Canada:

2012-341583 Hear the world research chair in music and emotional speech from Phonak. Zenodo, April 2018. DOI: 10.5281/zenodo.1188976. URL: https://doi.org/10.5281/zenodo.1188976.

[16] M. M. Duville, L. M. Alonso-Valerdi, and D. I. Ibarra-Zarate. The mexican emotional speech database (mesd): elaboration and assessment based on machine learning. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1644–1647, 2021. DOI: 10.1109/EMBC46164.2021.9629934.

[17] S. Latif, A. Qayyum, M. Usman, and J. Qadir. Cross lingual speech emotion recognition: urdu vs. western languages, 2020. arXiv: 1812.10411 [cs.CL].

[18] M. K. Pichora-Fuller and K. Dupuis. Toronto emotional speech set (TESS), version DRAFT VERSION, 2020. DOI: 10.5683/SP2/E8H2MF. URL: https://doi.org/10.5683/SP2/E8H2MF.

[19] S. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko. SemEval-2018 task 1: affect in tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics, June 2018. DOI: 10.18653/v1/S18-1001. URL: https://aclanthology.org/S18-1001.

[20] E. Saravia, H.-C. T. Liu, Y.-H. Huang, J. Wu, and Y.-S. Chen. CARER: contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics, October 2018. DOI: 10.18653/v1/D18-1404. URL: https://www.aclweb.org/anthology/D18-1404.