KAI: An AI-powered Chatbot To Support Therapy

Bachelor Thesis Project
Specilization in Computer Science

Mariama C. Djalo D.

Date: 23/01/2023
Director: Javier Béjar Alonso
Department: Computer Science
Degree: Bachelor of Computer Science
Center: FACULTAT D'INFORMÀTICA DE BARCELONA (FIB)
University: UNIVERSITAT POLITÈCNICA DE CATALUNYA (UPC) – BarcelonaTech

Contents

List of Figures

List of Tables

Acknowledgement

First of all, I want to thank my family for supporting me in my worst times and for sharing with me my best moments. Also, I want to thank my friends for accompanying me through this journey and for forming part of my good memories from College. Finally, I must express my gratitude to my director for guiding me during the development of the project, I really appreciate his support.

Abstract

This project attempts to bridge the huge gap between people who struggle with mental health and people who actually get treated with the design and implementation of Kai, an AI-powered Chatbot that supports Cognitive Behavioral Therapy (CBT).

CBT is based on identifying cognitive distortions (negative thoughts) and challenging them to improve mood and overall mental health. This was done by using Text Classification, a Natural Language Processing (NLP) technique to identify potential cognitive distortions in text.

During the project, a model experimentation was done to compare different supervised machine learning models in order to choose the best one for the text classification. The dataset needed to train the models was generated by manually giving labelled examples of the 15 major cognitive distortions. The model experimentation was entirely done in python.

Furthermore, the design of the dialogue flow of the chatbot was done following the CBT's guidelines and the implementation of the chatbot was done in Python using the TKinter in Framework for the interface. Finally, two test were made to check for the correct functioning of the chatbot.

KEYWORDS: Chatbot, CBT, Cognitive Distortions, Text Classification, NLP, Supervised Machine Learning.

Resum

Aquest projecte intenta reduir l'enorme bretxa existent entre les persones que pateixen problemes de salut mental i les que realment reben tractament amb el disseny i la implementació de Kai, un xatbot impulsat per IA que dóna suport a la teràpia cognitivoconductual (TCC).

La TCC es basa en la identificació de les distorsions cognitives (pensaments negatius) per qüestionar-les amb l'objectiu de millorar l'estat d'ànim i la salut mental. Per això es va utilitzar la Classificació de Text, una tècnica de Processament del Llenguatge Natural (PLN) per identificar potencials distorsions cognitives en el text.

Durant el projecte es va fer una experimentació de models per comparar diferents models supervisats d'aprenentatge automàtic per triar el millor per a la classificació de textos. El conjunt de dades necessari per entrenar els models es va generar proporcionant manualment exemples etiquetats de les 15 distorsions cognitives principals. L'experimentació del model es va fer íntegrament en python.

A més, el disseny del flux de diàleg del chatbot es va fer seguint les directrius del TCC i la implementació del chatbot es va fer a Python utilitzant TKinter, un Framework per a la interfície. Finalment, es van realitzar dos tests per comprovar el funcionament correcte del chatbot.

PARAULES CLAU: Chatbot, CBT, Distorsions Cognitives, Classificació de Text, NLP, Aprenentatge Automàtic Supervisat.

Resumen

Este proyecto intenta reducir la enorme brecha existente entre las personas que sufren problemas de salud mental y las que realmente reciben tratamiento con el diseño y la implementación de Kai, un chatbot impulsado por IA que da soporte a la terapia cognitivo-conductual (TCC).

La TCC se basa en la identificación de las distorsiones cognitivas (pensamientos negativos) para luego cuestionarlas con el objetivo de mejorar el estado de ánimo y la salud mental. Para ello se utilizó la Clasificación de Texto, una técnica de Procesamiento del Lenguaje Natural (PLN) para identificar potenciales distorsiones cognitivas en el texto.

Durante el proyecto se realizó una experimentación de modelos para comparar distintos modelos supervisados de aprendizaje automático con el fin de elegir el mejor para la clasificación de textos. El conjunto de datos necesario para entrenar los modelos se generó proporcionando manualmente ejemplos etiquetados de las 15 distorsiones cognitivas principales. La experimentación del modelo se realizó íntegramente en python.

Además, el diseño del flujo de diálogo del chatbot se hizo siguiendo las directrices del TCC y la implementación del chatbot se hizo en Python utilizando TKinter, un Framework para la interfaz. Por último, se realizaron dos tests para comprobar el correcto funcionamiento del chatbot.

PALABRAS CLAVE: Chatbot, CBT, Distorsiones Cognitivas, Clasificación de Texto, NLP, Aprendizaje Automático Supervisado.

1 Context

This is a Bachelor Project Thesis done at the Barcelona School of Informatics (FIB) under the supervision of Javier Béjar Alonso for the Computer Science Degree with a specialization in Computing.

1.1 Introduction

According to WHO [8], depression is a leading cause of disability and among people aged 15 to 29, suicide is the fourth most common cause of death. There is still a substantial disparity between those who require care and those who have access to care, despite the fact that many mental health illnesses may be adequately treated at low cost [9].

Therapeutic and mental health Chatbots are an intriguing way to bridge this gap. A chatbot is a computer system tool that is designed to simulate human communication.

On the other hand, it is widely known that a few sessions of cognitive-behavioral therapy (CBT) can be extremely beneficial in the treatment of anxiety and depression. However, many people do not have access to a CBT therapist because they cannot afford it, it is not covered by their insurance, or there are none nearby. It can also be difficult to go to therapy due to lack of time because of work or having to take care of kids for instance. However, in some cases, a therapist may not be required. There are numerous options for doing CBT without a therapist, such as self-help books and internet-based treatment, or by conducting your own research on the material. Self-directed CBT has been shown in numerous studies to be very effective [10]. This particular type of therapy promotes independence and self-therapy. The methodology of CBT is based on **identifying cognitive distortions**, known as negative thoughts and **challenging them** by replacing them for alternative and more positive thoughts in order to **improve mood and the overall mental health**.

The goal of this project is to combine the two previously mentioned methodologies to help people dealing with mental health issues. KAI, a Chatbot that provides therapy support in self-directed CBT, will be designed and implemented as part of the project. KAI will detect Cognitive Distortions in text using Text Classification, an NLP technique that will assist in the application of CBT.

This project aims to address the problem of a significant gap that exists between people who are struggling with mental health issues and need assistance and those who have access to mental health care. In this project, the design and development of an AI-powered chatbot that supports therapy by automatically detecting cognitive distortions is done in order to achieve this goal. The goal of this project is to train various supervised machine learning techniques for text classification in order to identify cognitive distortions from chatbot conversations and use them to aid in self-directed CBT.

The project is aimed at people who meet all of the following conditions:

- People who have mild to moderate symptoms of mental health issues and can function normally. [1]

[1] Those who are severely depressed or have severe mental health problems will most likely require one-on-one therapy with a professional

- People who do not have access to or prefer not to speak with a therapist due to privacy concerns and/or a fear of being judged as a result of the stigma associated with mental health issues.

- People who prefer to be autonomous and prefer to learn and treat themselves with self-directed therapy.

Bear in mind that **the chatbot is not a therapist and thus does not provide mental health diagnosis**; rather, the chatbot is a tool that assists with the automatic detection of potential cognitive distortions and guides you through the process of identifying and challenging them. In other words, KAI is a self-directed AI-based CBT tool that assists people in learning how to deal with cognitive distortions.

1.2 Cognitive Behavioral Therapy

One of the core concepts of the project is cognitive-behavioral therapy (CBT), thus, in this section is going ot be explained.

The goal of cognitive-behavioral therapy (CBT), a type of psychotherapy, is to help patients understand how their thoughts, beliefs, and attitudes affect how they feel and behave. The foundation of cognitive behavioral therapy (CBT) is the premise that our thoughts, feelings, and behaviors are connected, and that by altering our thoughts and beliefs, we can alter our behaviors and emotions.

1.2.1 Self-Directed Therapy

With little to no supervision from a therapist, patients can work independently during self-directed cognitive-behavioral therapy (CBT) to identify and change the harmful thoughts (cognitive distortions) and behaviors that contribute to their emotional and behavioral problems. Self-directed CBT may be a helpful treatment option for a patient who chooses to work independently or does not have access to a therapist.

Self-help tools like books, websites, or apps are widely used in self-directed CBT to guide the patient through the process of identifying and altering harmful thinking and behavior patterns.

1.2.2 Cognitive Model

According to Cully et al. [1] **"The cognitive model is a theoretical paradigm for explaining how thoughts, feelings, and behaviors are associated . Most individuals believe that situations give rise to their emotions"** as shown in Figure 1. **"The cognitive model challenges this subjective experience and suggests, instead, that it is the thoughts we have about situations that give rise to emotions"** as illustrated in Figure 2.

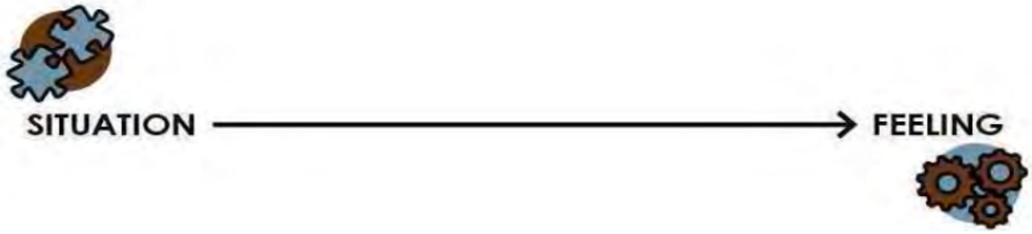

Figure 1: A drawing that depicts how a situation causes feelings, from the book *A Therapist's Guide to Brief Cognitive Behavioral Therapy* [1].

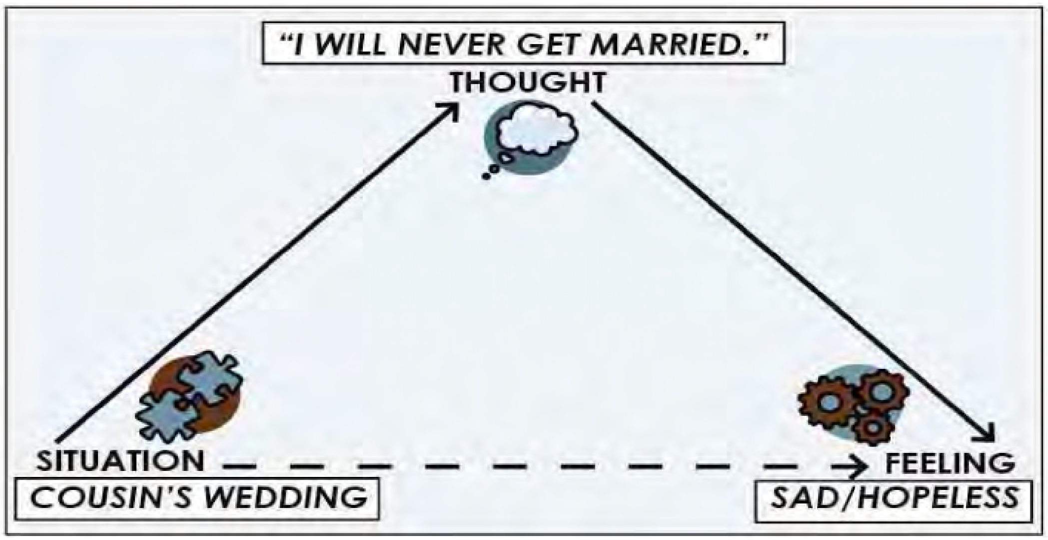

Figure 2: Diagram illustrating the structure of the Cognitive Model, from the book *A Therapist's Guide to Brief Cognitive Behavioral Therapy* [1].

1.2.3 Cognitive Distortions

Cognitive distortions are inaccurate, incorrect, or distorted ways of thinking that can have a negative impact on feelings and actions. These distortions alter how we perceive and understand events, which may lead us to form inaccurate or erroneous assumptions that have harmful consequences.

Typical examples of cognitive distortions are as follows:

- **Jumping to conclusions**: Drawing judgments without enough evidence. For example, *"She's not responding so she must be ignoring me or even worse, she must be mad at me"*
- **Mental filter**: Focusing only on the negative part of everything. For example, *"I did well on most of the exam, but I got one question wrong, so I must be a complete failure."*
- **Emotional reasoning**: Thinking that because you feel something it must be real. For example, *"I feel like a failure, so I must be a failure."*

1.3 Artificial Intelligence

Artificial intelligence (AI) is another important concept of the project and will be explained in this section. Artificial intelligence (AI) are computer systems that mimic human intelligence, for instance, learning, problem-solving, decision-making and understanding languages. AI applications include face recognition, chatbots, and self-driving cars.

1.3.1 NLP

Natural language processing (NLP), as illustrated in Figure 3, is a subfield of artificial intelligence. NLP focuses on teaching computers to comprehend text and spoken language in the same way that humans do.

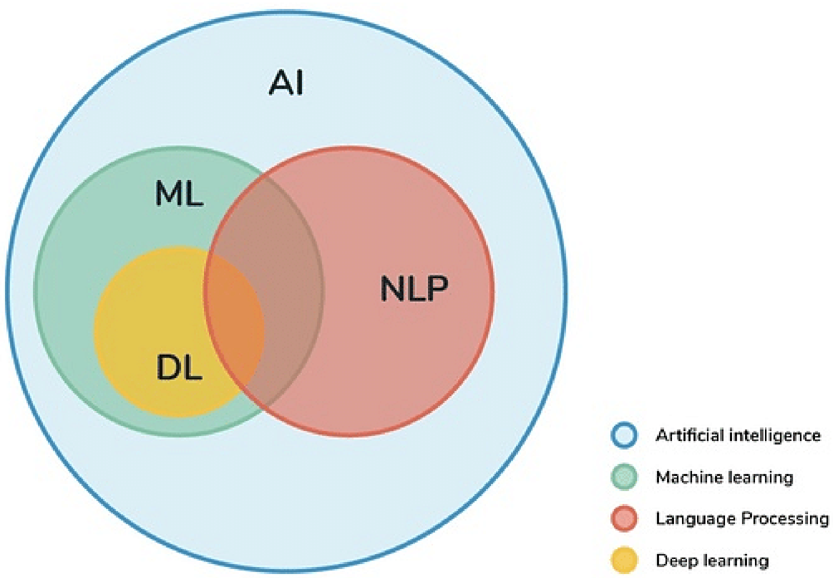

Figure 3: A diagram showing the different subfields of Artificial Intelligence, from *European-Valley* [2], a website.

1.3.2 Machine Learning

Machine Learning, as shown in Figure 3, is another subfield of AI based on the idea that computers are capable of learning from data, recognizing patterns and making decisions with little to no human intervention.

Machine learning algorithms are used to build models that can make decisions without being coded to perform an specific task, in other words, autonomous systems. These algorithms are able to learn from data, and improve over time, by identifying patterns and relationships within the data.

Supervised learning and unsupervised learning are the two main types of machine learning. In supervised learning, the model is trained on labeled data, where the correct output is provided for each example in the training set. The goal is to learn a function that can predict the output for a new input.

In unsupervised learning, the model is not given labeled training examples; instead, it must use methods like clustering to determine the underlying structure of the data.

Features and Feature Selection

In machine learning, features refer to the input variables or characteristics that are used to describe and predict the output or target variable. Feature selection is the process of selecting a subset of the most relevant and informative features for building a model.

Overfitting in Machine learning

Overfitting occurs when a model learns the information and noise in the training data to the extent that it negatively affects the model's performance on new data. This indicates that the machine learns concepts from the noise or random fluctuations in the training data. These concepts don't apply to new data, which poses a difficulty for the models' capacity to generalize.

Nonparametric and nonlinear models, which have more flexibility while learning a target function, are more susceptible to overfitting. As a result, a lot of nonparametric machine learning algorithms additionally incorporate parameters or methods to restrict and limit the amount of information the model may learn.

High Variance

In machine learning, "high variance" refers to a model's propensity to have a significant difference between its training error and its testing error. This can occur when the model is overly sensitive to the specific details of the training data and is not able to generalize well to unseen data.

High variance might be problematic since it indicates that the model will probably perform badly when applied to new, unseen data. Additionally, it can be a sign of overfitting, in which case the model learns the random fluctuations and noise seen in the training data rather than the underlying pattern.

Bias

In machine learning, bias is the systematic discrimination or inaccuracy that appears in a model's predictions. A number of factors, such as the use of biased training data or the model's construction itself, can lead to bias.

Because bias can result in unfair or erroneous predictions that might reinforce or magnify already-existing societal inequalities, bias can have major negative effects. For instance, a biased employment model might disproportionately reject specific minority groups, while a biased lending model might unfairly refuse loans to particular borrowers.

To make sure that the model's predictions are accurate and fair, bias in machine learning must be carefully considered and addressed. This may require using a varied and representative training dataset, minimizing bias in the model using strategies like regularization, and regularly evaluating the model's performance on different subgroups.

Data Crowdsourcing

Data crowdsourcing is the process of gathering data from a big group of people, frequently via an internet platform. This may be a useful method for quickly and inexpensively gathering a lot of data.

1.3.3 Text Classification

Text Classification is a supervised machine learning technique in NLP used to classify text into different categories. Common types of Text Classification are Sentiment Analysis and Spam Email detector.

As shown in Figure 4, the Text Classification Workflow begins with gathering data (obtaining a dataset) and continues with data exploration using descriptive analytics. The next step is to prepare the data. Since I'll be working with texts and ML models don't understand them, I'll need to do Text Vectorisation, which is the process of converting text into numerical representation.

Following that, the model is built, trained with the dataset, and evaluated to see how well it performs. In my case, I'll train several models before selecting the best one.

Finally, after Hyperparameters Tuning for improved performance, the model is deployed. Hyperparameters are the parameters that define the model architecture, and Hyperparameter Tuning is the process of searching for the best model architecture. The maximum depth is an example of a Hyperparamter in decision trees (a ML Model).

Figure 4: ML model workflow overview, from *Google Machine Learning Education* [3], a website.

1.4 Justification

The use of AI in mental healthcare and psychiatry, particularly therapeutic chatbots, is still in its early stages, with limited research data and datasets available to fully explore the field's true potential.

Shickel et al. [11] published the most relevant research in 2019. The authors used supervised machine learning techniques such as SVM, XGBoost, and RNN to find cognitive distortions in text automatically. The authors obtained a weighted F1 scores of 0.68.

There are currently no datasets containing labeled text sections with distortions that are available to the general public because the detection of cognitive distortions is a novel machine learning task. They gathered information using a real-world online therapy program and crowdsourcing in order to resolve this difficult problem.

Toledo et al. [12], on the other hand, took a different and very innovative approach: they developed cognitive distortion responses to support CBT interactions. Which is extremely important because Cognitive Behavioural Therapy's (CBT's) core idea is the ability to change distorted or negative beliefs (cognitive distortions) into more realistic alternatives (positive thoughts). The authors used Transformers learners to generate the responses.

It's also worth mentioning that there are self-help apps and chatbots for mental health, many of which use CBT, on the market, including in Spain[2]. Wysa and Woebot are the most well-known.

As mentioned before, there are currently no publicly available datasets. Crouwdsourcing and data generation are two options for resolving this challenging issue. Since Crowdsourcing is a paid service, I will opt for the second option, in other words: I will create the dataset myself. The process will be completed by providing sufficient examples of the 15 major cognitive distortions. To ensure that the data is accurate, examples will be drawn or be inspired from psychological books, websites, and articles specializing in psychology.

Unfortunately there is no public information available on how to design a CBT chatbot. Therefore, the design will be done from scratch with the aid of a very helpful and introductory manual book to CBT [1] that will help have a better understanding about the structure of cognitive behavioural therapy session.

1.5 Scope

The project's objectives, functional requirements, and potential risks and obstacles are all discussed in this section.

1.5.1 Objectives

The main goal of this project is to design and implement an AI-powered Chatbot that supports therapy to assist people suffering from mental health issues such as anxiety or depression. Kai will use CBT technique and with the help of AI he will automatically detect cognitive distortions. To achieve this goal, the project has been divided into several subobjective:

Theoretical part

1. Investigate CBT to gain a thorough understanding of how it works.
2. Design the chatbot's conversation structure and dialogue flow.
3. Investigate the best supervised machine learning techniques for text classification.

[2] This fact confirms and clarifies that it is perfectly legal to make **support therapy chatbots**

Practical Part

1. Generate a suitable dataset to train the machines.
2. Train different supervised machine learning models.
3. With the help of metrics, choose the best model.
4. Implement the chatbot.

1.6 Requirements

There are some requirements that must be met in order to ensure the final project's quality:

- **Data pre-processing.** Data pre-processing is essential for ensuring that the models perform optimally.
- **Hyperparameter Tuning.** Understanding how different algorithms work is essential for knowing how to make good Hyperparameter Tuning decisions to achieve the best results.
- **Optimization of the code.** Optimize the code for all imputation methods as well to improve efficiency.
- **Avoid bias.** Avoiding bias is key to ensure that the models are accurate.
- **Good programming practices.** Using good programming practices such as a readable style, good comments, and as little complexity as possible.

1.7 Risks

There may be some risks that prevent the project from progressing smoothly:

- **Not being able to generate an appropriate or a representative dataset to train models.** The main risk would be not being able to find/generate a suitable (without bias) dataset to train the models, which would prevent the project from progressing.
- **Deadline of the project.** The project also has a deadline for completion that must be met. This forces to make difficult decisions. As a result, strong organizational skills and the ability to meet deadlines are essential for finishing the project on time. Some libraries, on the other hand, have flaws.
- **Bugs in libraries.** Some libraries may have bugs in certain functions, resulting in incorrect code.

1.8 Methodology

In order to have more flexibility, I will use a hybrid of Waterfall and Agile workflow methodology for this project. Waterfall is a project management methodology that is based on a sequential design process. Agile is a methodology that prioritizes development through evolution. This method enables sprint work and the resolution of issues that arise during iterations. The Agile-Waterfall hybrid method combines the best features of both methods: Agile allows you to check for bugs, test the code, and correct it in a progressive way without having to wait until the entire implementation is completed. Waterfall, on the other hand, allows you to keep track of all the dependencies between tasks to better organize the project.

The Kanban framework which falls under the Agile methodology will be used. Since the 1950s, the Japanese phrase ”kanban,” which means ”visual board” or a ”sign,” has been used to refer to a process definition. Toyota invented it and used it as the first just-in-time factory scheduling system. The ”Kanban Method,” which was initially defined in 2007, is known and connected with the capitalized term ”Kanban,” on the other hand.

Kanban boards are used to efficiently show and control the workflows. The essential elements are:

- **Kanban Cards** are used to express tasks visually. Each card contains details on the work and its progress, including the due date, the person assigned to it, the description, etc.

- **Kanban Columns**: On the board, each column corresponds to a distinct step of your operation. The workflow is applied to the cards till they are finished completely.

In my case, I'll have four columns.

1. **To Do:** composed by all of the tasks that haven't been started yet.

2. **In progress:** composed by all of the tasks that are still in progress.

3. **Tested:** composed by all of the tasks that have already been completed but need to be tested to ensure they work properly.

4. **Completed:** composed by all completed and tested tasks.

There are many project management tools that follow the kanban methodology, but I'm going to use Trello because I believe it's the better option for small teams or one-person teams, such as freelancers (which is my case).

On the other hand, a Gantt chart with a Waterfall workflow will also be used to keep track of the dependencies and the required time for each task.

I'll use a Github repository as a version control tool to make sure I can restore earlier versions in the event of serious errors because it's securely kept in the cloud.

I'm planning to use the GitHub Flow as my Git branching technique. Its branches are organized into a main branch where the code that is ready for production is kept and additional branches, referred to as feature branches, where work on new features and bug fixes is done and then merged back into the main branch. Smaller teams, like mine, as I've already mentioned, benefit most from this approach.

For the practical part, the **cross-validation** method will be used to choose the optimum **Hyper-parameters** for the models and also to check their **performance** and if there is **bias**. I'll also set the random state to an integer to prevent having different outcomes each time I run the model.

Last but not least, whenever I have queries or run into problems, I shall email or meet with my tutor online. Extraordinary in-person meetings will be scheduled if I encounter any serious issues with my project or if I believe that communicating with my tutor in person will be more convenient.

1.9 Stakeholders

In this section the stakeholders who will benefits from the completion of this project will be enumerated.

The project's completion is important not only for people suffering from mental illnesses, but also for hospitals, clinics, educational institutions, and communities/groups in general. They could use the chatbot to treat and improve the mental health of patients/community members, thereby improving people's overall well-being.

- The project's completion is specially important for people struggling with mental health.

- Hospitals, clinics, educational institutions, and communities/groups in general also benefit from this project. They could use the chatbot to treat and improve the mental health of patients/community members, thereby improving people's overall well-being

2 Project Planning

The project will last approximately 579 hours spread over 126 days, beginning on September 20th, 2022 and ending on January 23rd, 2022. Since the date for the project defense has not yet been determined, the previous deadline is the earliest we can have. It is planned to work an average of 5 hours per day, but some flexibility may be required due to exams or personal issues.

This section begins with a task description, followed by the resources required for the project's development, and finally by an explanation of risk management. Furthermore, Table 1 summarizes all of the defined tasks, as well as their dependencies and required resources, and Figure 5 captures the project schedule.

2.1 Description of tasks

The identification and description of the tasks that will be completed during the course of the project are presented in this section. It is given an estimate of the time needed for each task in hours, as well as a description of the logical sequence and dependencies between them. The tasks are divided into the following groups: **Project Management, Project Research, Project Theory, Data Generation, Project Experimentation, Project Development, Project Documentation and Thesis Defense Preparation.**

2.1.1 Project Management

Project management is most likely one of the project's pillars. It defines the scope of the project, the tasks and their planning, as well as the budget and sustainability.

- **PM1 - ICT tools for project and team management.**
  - **Description:** To support the project's development, the best technology, devices, and concepts that fit the nature of our project are required. To accomplish this, it is necessary to conduct research on various types of software for various tasks.
  - **Resources:** PC with internet acces.
  - **Approximate duration:** 1 hour.
- **PM2 - Context and Scope.**
  - **Description:** The project's scope, as well as its contextualization, are defined. The general goal of the project, its justification, developments, and tools are all discussed in this section.
  - **Resources:** This part requires a PC with internet connection, Overleaf to document and both The GEP [3] Tutor and the Tutor of the project for the feedback.
  - **Approximate duration** 35 hours.

[3] GEP is a course that must be passed before submitting the final version of the thesis. It stands for "Gestió de Projectes" which translates to Project Management.

- **PM3 - Time planning.**
  - **Description:** It is critical to plan ahead of time to ensure that the deadline is met. A good plan can help us identify which tasks require more attention than others and which are critical.
  - **Resources:** Making the Time Planning requires a PC, Overleaf, The GEP Tutor and the Tutor and TeamGantt to make the Gantt chart.
  - **Approximate duration:** 30 hours.
- **PM4 - Budget and sustainability.**
  - **Description:** The main objectives of this assignment are to develop a budget and assess the project's sustainability. This is important to know in order to calculate the project's overall cost and its development impact.
  - **Resources:** This part needs a PC, Overleaf, The GEP Tutor and the Tutor as resources.
  - **Approximate duration:** 30 hours.
- **PM5 - Meetings.**
  - **Description:** Meetings with the project's tutor will be scheduled as needed (E.g. when doubts or critical problems that impede the proper development of the project arise. I have added a "reserved time" for the meetings to make a better estimation of the total hours required for the project.
  - **Resources:** Tutor.
  - **Approximate duration:** 1 hour a week (in total, 18 hours).

2.1.2 Project Research

This part has been divided into the following tasks :

- **PR1 - Psychology research.**
  - **Description:** This project's methodology is based on psychology, specifically Cognitive Behavioral Therapy. Before embarking on the implementation phase, thorough research in CBT is essential.
  - **Resources:** PC, Books, Research Papers, Articles.
  - **Approximate duration:** 10 hours.
- **PR2 - ML research.**
  - **Description:** Documentation on various types of supervised machine learning models, as well as the statistics behind them, is also highly needed.
  - **Resources:** PC, Books, Research Papers, Articles.
  - **Approximate duration:** 10 hours.

2.1.3 Project Theory

In the theoretical part, I will study the structure of a CBT Therapy session in order to properly design the dialogue flow. Furthermore, I will consider what are the best ML models for both text classification and small datasets [4].

This part is divided into the following tasks:

- **PT1 - Design.**
  - **Description:** Design of the structure and dialogue flow of the chatbot.
  - **Resources:** PC, Books, Research Papers, Articles.
  - **Approximate duration:** 10 hours.
- **PT2 - Choose.**
  - **Description:** Choose between the top 3-5 supervised ML models for text classification and small datasets.
  - **Resources:** PC, Books, Research Papers, Articles.
  - **Approximate duration:** 10 hours.
- **PT3 - Select the Hyperparameters.**
  - **Description:** Select the Hyperparameters of every model to optimize.
  - **Resources:** PC, Books, Research Papers, Articles.
  - **Approximate duration:** 10 hours.

2.1.4 Data Generation

- **Description:** It is necessary to **generate the dataset** before beginning the experimentation with the models to determine which is the best. As previously stated, the generation will be created by using examples or finding inspiration on the internet, books, and articles specialized in psychology. The dataset will include examples of the 15 major cognitive distortions shown in Figure 11.
- **Resources:** PC, Books, Research Papers, Articles.
- **Approximate duration:** 14 hours.

[4] Since I will be creating the data, the dataset will most likely be quite small (less than 1000 examples of cognitive distortions).

2.1.5 Project Experimentation

In the experimentation section, the ml model workflow is completed as described in the previous assignment, and the best model is selected after analyzing the metrics.

In summary, the tasks are divided into the following:

- **PE1 - Apply the workflow.** This part will require 16 hours.
- **PE2 - Hyperparameters tuning.** Experiment with every model optimizing them with the aid Hyperparameters tuning selected in the Project Theory part (thus there is a dependency). This part will require 8 hours.
- **PE3 - Choose the best ml model.** Analyse the performance of every model and choose the best one for the text classification. This part will require 2 hours.

All these tasks will be done in Colaboratory, best known as ”Colab”. This tool is a product from **Google Research**. **Colab** is the best tool for machine learning, data analysis, and education since it enables anyone to create and execute arbitrary Python code in the browser. Google Drive is where Colab notebooks are kept, making it a secure cloud storage option. Furthermore, a PC, books, research papers, articles, programming languages and Github will be needed.

2.1.6 Project Development

The Project Development takes place after designing the structure and dialogue flow of the chatbot and consists on the implementation of a telegram chatbot.

The tasks are organized as follows:

- **PDEV1 - Implementation.**
  - **Description:** Implement the telegram chatbot.
  - **Resources:** PC, Github, Colab and programming languages.
  - **Approximate duration:** 200 hours.
- **PDEV2 - Testing.**
  - **Description:** Test the correct functioning of the telegram bot. This will be done during and after the implementation of the apps in order to make sure all parts work correctly on time.
  - **Resources:** PC, Github, Colab and programming languages.
  - **Approximate duration:** 60 hours.

2.1.7 Project Documentation

To avoid having to do everything at the end, the project documentation will be completed concurrently with the project's development (after the research part is done). For these task we need a PC, Overleaf/Texifier and Trello to keep track of the progress. The Project Documentation has been broken down into the following tasks:

- **PDOC1 - Annotation of events:** Annotation of all the events that are done during the project development. This task will be done intermittently and will aproximately require 10 hours.
- **PDOC2 - Revision of the events:** Once the project development is done is time to check all the documentation done during the project to better organize the ideas, correct changes and structure the final document correctly. This task will require 20 hours.
- **PDOC3 - Write Final Document:** After PDOC2 is done the writing of the final documentation begins. This task will require approximately 60 hours.

2.1.8 Thesis Defense Preparation

Finally, after the project documentation is completed, the oral defense preparation begins. To do so, it is necessary to practice and prepare for potential tribunal questions. This will require 25 hours.

2.2 Risk management: alternative plans

During the course of the project, difficulties that could jeopardize the project's proper progress may arise. All of the previously introduced potential problems in section 1.7 will be addressed in this section by introducing new tasks and accommodating the planning. A level of risk is also added to each of them.

- **Not being able to generate an appropriate or a representative dataset to train the models [Extreme Risk]**. Bias may be introduced into the data by providing examples by myself because I am forcing the examples to fit the criteria for cognitive distortions, which may or may not reflect how cognitive distortions occur in the real world. CD *"in the wild"* may be more subtle or more than one can appear in a thought/phrase. If that were to occur these are the steps that would be followed:
  - **Scrapping the internet or obtaining a dataset from a public mental health/therapy forum.** In my case, I would use a public dataset obtained from Reddit (specifically, from mental health subreddits) [13]. This method would yield "real-world" examples. Unfortunately, Manual labeling of the dataset into the 15 major cognitive distortions would be required.
  - **Crowdsourcing.** People will give examples that fit the CD criteria, so there may still be some bias. However, it would be significantly more varied than giving examples by myself in this case because people from different parts of the world

would give examples (that may or may not apply to their real-life situation) that would most likely differ from mine due to differences in backgrounds, for example.

  - **Resources to reuse:** PC, programming languages, and TeamGantt.
  - **Estimated delay:** between 1-2 weeks.

- **Deadline of the project [High Risk]**. It could be caused by an accurate preliminary estimation of the tasks and their duration, which is completely normal because we do it before we begin. It is important to plan ahead of time, but it is also important to be flexible and adapt to changing circumstances. If that were to occur these are the steps that would be followed:
  - **Replanning in a more advanced of the project.** We can easily solve this problem by replanning in a more advanced part of the project, allowing us to do it more accurately.
  - **Increasing the hours dedicated to the project.** If, despite planning, it is still difficult to meet the deadline, it may be resolved by increasing the number of hours dedicated to the project as a last resort.
  - **Resources to reuse:** PC and TeamGantt.
  - **Estimated delay:** 1-2 weeks.

- **Bugs in libraries [Medium Risk]**. Third-party libraries will be used during project development, and they may contain bugs. Waiting until the library is updated, which should hopefully fix the bug, is one possible solution. However, due to the tight deadline, this option is out of the question. As a result, coding the function from scratch and testing its correct operation would be required, increasing the overall duration of the project.
  - **Resorces to reuse:** PC, TeamGantt and programming languages.
  - **Estimated delay:** 1-2 weeks.

ID	Name	Time (h)	Dependencies	Resources
PM	**Project Management**	**114**		
PM1	ICT tools for project and team management	1		PC
PM2	Context and Scope	35		PC, Overleaf, GEPT, T
PM3	Time planning	30	PM2	PC, Overleaf, GEPT, T, TeamGantt
PM4	Budget and sustainability	30	PM3	PC, Overleaf, GEPT, T
PM5	Meetings	18		T
PR	**Project Research**	**20**		PC, Books, Research Papers, Articles
PR1	Psychology Research	10		
PR2	ML Research	10		
PT	**Project Theory**	**30**		
PT1	Design Chatbot	10	PR1	PC, Books, Research Papers, Articles
PT2	Choose supervised ML models	10	PR2	PC, Books, Research Papers, Articles
PT3	Select the Hyperparameters to optimize	10	PT2	PC, Books, Research Papers, Articles
DG	**Data Generation**	**14**		PC, Books, Research Papers, Articles
PE	**Project Experimentation**	**26**		
PE1	Apply the Workflow	16	DG	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PE2	Hyperparameters Tuning	8	PT2, PE1	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PE3	Performance analysis of every model	2	PE2	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PDEV	**Project Development**	**260**		
PDEV1	Telegram bot Implementation	200	PT1	PC, Programming languages, Github, Colab
PDEV2	Testing	60	PT1	PC, Programming languages, Github, Colab
PDOC	**Project Documentation**	**90**		
PDOC1	Annotation of events	10	PR	PC, Textit, Trello
PDOC2	Revision of the events	20	PR	PC, Textit, Trello
PDOC3	Write final documentation	60	PR	PC, Textit, Trello
TDP	**Thesis Defense Preparation**	**25**	**PDOC**	PC, Textit, results
Total		**579**		

Table 1: Task Table containing a summary of all task information. T and GEPT means Tutor and GEP Tutor respectively. [Own Creation]

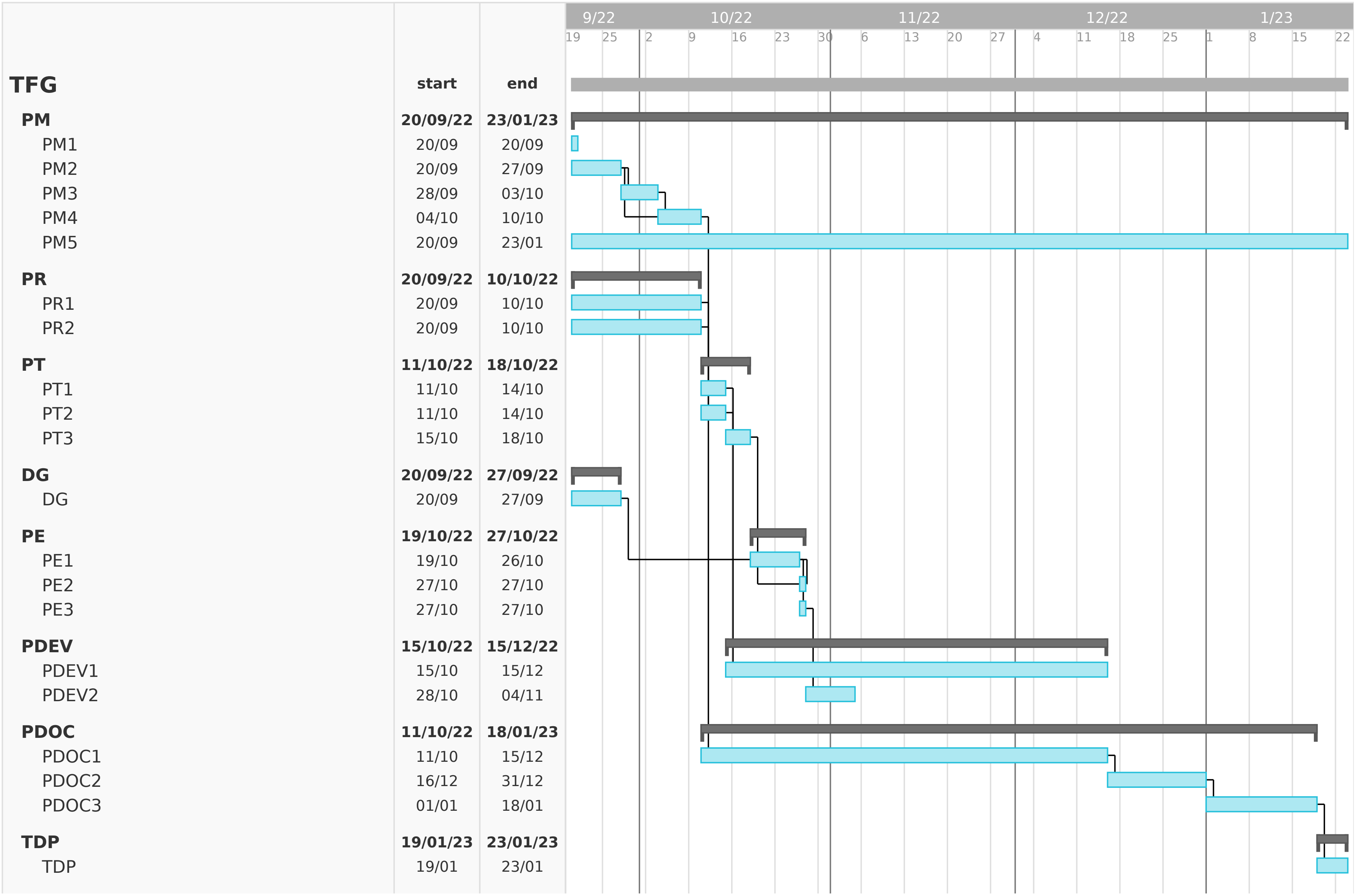

Figure 5: Gantt Chart illustrating the project's schedule following a Waterfall model. [Own Creation]

2.3 Budget

In this section the economic cost of the project is discussed. First, the staff cost is described and analysed, then the generic and indirect costs are also calculated. Furthermore, the mechanism for controlling potential budget deviations is also explained. Finally, in Figure 3 we can see that the budget estimation is **17066,08€**.

2.3.1 Costs Per Activity

To accurately estimate the project's costs and create a budget, we must consider all of the resources required. Human resources is one of them. Even though the project is going to be done only by my with the guidance of my tutors, 7 roles are created to better simulate the required human resources to develop the project in order to better estimate the cost by task. The following is a description of the various responsibilities of each role:

- **Project Manager.** The project manager is in charge of the project's planning and development; in other words, the project manager oversees the project's progress.
- **Software engineer.** The software engineers implements the chatbot.
- **Tester.** The tester is in charge of verifying that the implementation is correct.
- **Research ML.** The researcher is responsible for investigating the best supervised machine learning models for the project and selecting the best hyperparameters for optimization.
- **Research psychologist** is responsible of designing the proper dialogue flow structure following the cbt method.
- **Technical writer** is in charge of documenting the project and presenting.
- **ML engineer** is responsible of implementing, tuning and analysing the models to choose the best one.

The Project Managements roles are going to be played by the tutors and the rest of roles by me.

In this section it is computed the Total Personnel Cost Per Activity (CPA). Each task or activity (previously defined in 2.1) is associated with the staff cost who are involved in that task. In this project there are 7 roles, each one with a different hourly salary which translates into **cost per hour** shown in Figure 2.

Role	Gross Annual Salary (€)	Price per hour (€)
Project Manager	52899,6	25,425
Software Engineer	46556,9	22,375
Tester	39533	19
Research ML	47239,4	22,7
Research psychologist	71436,3	34,35
Technical Writer	41600	20
ML engineer	47239,4	22,7

Table 2: Salary of the different roles extracted from PayScale, a compensation software company [7] multiplied by 1.35 to include the cost of social security [Own creation].

The computation of CPA is done by multiplying the hours required per task/activity with the cost per hour of the role that is involved in the activity. The total CPA is the sum of the CPA of every task of the Gantt Chart. As shown in Figure 3, the total cost of recruitment (CPA) is **13366,05€**.

2.3.2 Generic Costs

There are many resources that aren't directly tied to a task: the generic costs. To calculate the generic costs we need to take into account the **amortisation** of the resources used. In this project, all the software products are free so we are going to focus on the calculation of the hardware costs. I will be working 5 hours a day on average during 126 days. The computation of the amortisation is done with formula 1.

$$Amortisation(€) = Resource\ Price \cdot \frac{1}{Years\ of\ Use} \cdot \frac{1}{Days\ of\ Work} \cdot \frac{1}{Hours\ per\ Day} \cdot Hours\ Used \tag{1}$$

The indirect costs are identified to make the budget more realistic. Since I'll be working from home for the project (unless an extraordinary in-person meeting with the tutor is required), the transportation cost is zero. On the other hand, internet costs around 70€ per month, and electricity costs 100€ per month. The total Generic Cost, as shown in Figure 3, is **1223,11€**.

2.3.3 Contingency

Unexpected events are common during the development of a project, and one must plan ahead to account for them. As a result, a contingency plan is created in order to avoid potential delays during the planning process. Since contingency margins in the IT sector typically range from 10% to 20%, I decided to have a 15% contingency margin for this project, which amounts to **2188,37€**.

2.3.4 Incidental Costs

Incidental costs define all potential risks that could cause project delays. The most extreme risk of the project in this case is detecting bias in the machine learning models and thus having to generate more data alternatively, as explained in previous sections. The project is delayed as a result of this risk. Total Incidental Costs are **288,54€**.

2.3.5 Management control

The budget control mechanisms are discussed in this section. Additionally, the control indicators that help monitor cost variances throughout the project's development are defined.

While doing each planned task, the deviation from the estimated cost is calculated with the Formula 2.

$$Deviation(€) = Cost_{Estimated} - Cost_{Real} \tag{2}$$

If the deviation is negative, part of the contingency fund must reallocated in order to cover the deviation. In the positive case, it means there has been a overestimation of cost and reallocation of the extra money to incidents would be more productive.

Activity	Amount (€)	Observations
PM1 - ICT tools for project and team management	25,43	Project Manager , 1 hour
PM2 - Context and Scope	889,88	Project Manager , 35 hours
PM3 - Time planning	762,75	Project Manager , 30 hours
PM4 - Budget and sustainability	762,75	Project Manager , 30 hours
PM5 - Meetings	457,65	Project Manager , 18 hours
PR1 - Psychology Research	343,50	Research psychologist, 10 hours
PR2 - ML Research	340,50	Research ML, 10 hours
PT1 - Design Chatbot	343,50	Research psychologist, 10 hours
PT2 - Choose supervised ML models	227,00	Research ML, 10 hours
PT3 - Select the Hyperparameters to optimize	227,00	Research ML, 10 hours
DG - Data Generation	480,90	Research psychologist, 14 hours
PE1 - Apply the Workflow	363,20	ML engineer, 16 hours
PE2 - Hyperparameters Tuning	181,60	ML engineer, 8 hours
PE3 - Performance analysis of every model	45,40	ML engineer, 2 hours
PDEV1 - Telegram bot Implementation	4475,00	Software Engineer, 200 hours
PDEV2 - Testing	1140,00	Tester, 60 hours
PDOC1 - Annotation of events	200,00	Technical Writer, 10 hours
PDOC2 - Revision of the events	400,00	Technical Writer, 20 hours
PDOC3 - Write final documentation	1200,00	Technical Writer, 60 hours
TDP - Thesis Defense Preparation	500,00	Technical Writer, 25 hours
Total CPA (Cost Per Activity)	**13366,05**	Total personnel costs by activty (Gantt actvities)
Hardware		
Laptop	220,57	Mackbook Air 2017, Purchase Price: 1200e
Peripheral devices	122,54	Display + mouse + keyboard, Purchase Price: 400e
Software		
Overleaf	0,00	Free to use
Google sheets	0,00	Free to use
TeamGantt	0,00	Free to use
Colab	0,00	Free to use
GitHub	0,00	Free to use
Space		
Electricity	400,00	100€/month x 4 months (duration of project)
Furniture	200,00	Table + Chair
Internet	280,00	70€/month x 4 months (duration of project)
Transport	0,00	Work from home
Total GC (Cost computed Generically)	**1223,11**	
Total Cost (Total CPA + Total GC)	**14589,16**	
Contingency	2188,37	Contingency margin = 15%
Total DC (direct cost) + IC (indirect cost) + Contingency	**16777,54**	
Data Generation Delay (1 Week)	240,45	Cost: Research psychologist, 14 hours. Risk: 50%
Data Generation Delay (2 Week)	48,09	Cost: Research psychologist, 14 hours. Risk: 10%
Total incidentals (or unforseen costs)	**288,54**	
TOTAL	**17066,08**	

Table 3: Budget Structure of the project [Own creation]

2.4 Deviations

The project's methodology hasn't changed; the hybrid mode between waterfall and agile technique is well suited for the project, and it's because of this that the previously mentioned deviations haven't had a significant impact on the project's proper development.

The gantt chart, which employs a waterfall methodology, was used throughout the project's development to determine the dependencies and the order in which to strategize the tasks. On the other hand, the project's coding and testing phases used an agile methodology with kanban boards to keep track of all the tasks.

As can be seen in Table 4, there were two significant changes: one that affected the project development and the other that affected the project documentation. Both changes had an impact on the budget. The changes will be described in the section that follows.

ID	Name	Time (h)	Dependencies	Resources
PM	**Project Management**	**114**		
PM1	ICT tools for project and team management	1		PC
PM2	Context and Scope	35		PC, Overleaf, GEPT, T
PM3	Time planning	30	PM2	PC, Overleaf, GEPT, T, TeamGantt
PM4	Budget and sustainability	30	PM3	PC, Overleaf, GEPT, T
PM5	Meetings	18		T
PR	**Project Research**	**20**		PC, Books, Research Papers, Articles
PR1	Psychology Research	10		
PR2	ML Research	10		
PT	**Project Theory**	**30**		
PT1	Design Chatbot	10	PR1	PC, Books, Research Papers, Articles
PT2	Choose supervised ML models	10	PR2	PC, Books, Research Papers, Articles
PT3	Select the Hyperparameters to optimize	10	PT2	PC, Books, Research Papers, Articles
DG	**Data Generation**	**14**		PC, Books, Research Papers, Articles
PE	**Project Experimentation**	**26**		
PE1	Apply the Workflow	16	DG	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PE2	Hyperparameters Tuning	8	PT2, PE1	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PE3	Performance analysis of every model	2	PE2	PC, Books, Research Papers, Articles, Programming languages, Github, Colab
PDEV	**Project Development**	**206**		
PDEV1	Chatbot Implementation	146	PT1	PC, Programming languages, Github, Jupyter Notebook, VS Code
PDEV2	Testing	60	PT1	PC, Programming languages, Github, Jupyter Notebook, VS Code
PDOC	**Project Documentation**	**105**		
PDOC1	Monitoring Report	15		PC, Overleaf, Trello, T
PDOC2	Annotation of events	10	PR	PC, Textit, Trello
PDOC3	Revision of the events	20	PDOC2	PC, Textit, Trello
PDOC4	Write final documentation	60	PDOC3	PC, Textit, Trello
TDP	**Thesis Defense Preparation**	**25**	**PDOC**	PC, Textit, results
Total		**540**		

Table 4: Final version of the task table [Own creation]

2.4.1 Deviations in the project development

After revising the documentation of the telegram app API to develop the chatbot, I opted to use another alternative: BotUI a javascript framework. The reason to this was that I found the Telegram API documentation rather unclear and not easy to use for the functionalities needed for the chatbot.

Unfortunately, there was a problem with the API connecting the front end and the model, and I had to make another change because of the time constraints. The implementation was done in a Jupyter notebook and TKinter, a python framework to develop the interface.

Additionally, the time required to do the project development was reduced from 200 hours to 146 hours.

2.4.2 Deviations in the project documentation

Writing the monitoring report wasn't factored into the initial project planning. As a result, the monitoring report has been included in the project's final planning. This modification affected both the computation of the budget, as shown in Table 5, and the projected number of hours required to complete the project, as shown in Table 4.

2.4.3 Deviations in the budget

Using the Formula 2, the total deviation cost is **17066,08 - 15995,01 = 1071.07€**. Since the deviation is positive, this amount is the saved amount of budget.

Activity	Amount (€)	Observations
PM1 - ICT tools for project and team management	25,43	Project Manager , 1 hour
PM2 - Context and Scope	889,88	Project Manager , 35 hours
PM3 - Time planning	762,75	Project Manager , 30 hours
PM4 - Budget and sustainability	762,75	Project Manager , 30 hours
PM5 - Meetings	457,65	Project Manager , 18 hours
PR1 - Psychology Research	343,50	Research psychologist, 10 hours
PR2 - ML Research	340,50	Research ML, 10 hours
PT1 - Design Chatbot	343,50	Research psychologist, 10 hours
PT2 - Choose supervised ML models	227,00	Research ML, 10 hours
PT3 - Select the Hyperparameters to optimize	227,00	Research ML, 10 hours
DG - Data Generation	480,90	Research psychologist, 14 hours
PE1 - Apply the Workflow	363,20	ML engineer, 16 hours
PE2 - Hyperparameters Tuning	181,60	ML engineer, 8 hours
PE3 - Performance analysis of every model	45,40	ML engineer, 2 hours
PDEV1 - Chatbot Implementation	3266,75	Software Engineer, 146 hours
PDEV2 - Testing	1140,00	Tester, 60 hours
PDOC1 - Monitoring Report	300,00	Technical Writer, 15 hours
PDOC2 - Annotation of events	200,00	Technical Writer, 10 hours
PDOC3 - Revision of the events	400,00	Technical Writer, 20 hours
PDOC4 - Write final documentation	1200,00	Technical Writer, 60 hours
TDP - Thesis Defense Preparation	500,00	Technical Writer, 25 hours
Total CPA (Cost Per Activity)	**12457,80**	Total personnel costs by activty (Gantt actvities)
Hardware		
Laptop	205,71	Mackbook Air 2017, Purchase Price: 1200e
Peripheral devices	114,29	Display + mouse + keyboard, Purchase Price: 400e
Software		
Overleaf	0,00	Free to use
Google sheets	0,00	Free to use
TeamGantt	0,00	Free to use
Colab	0,00	Free to use
GitHub	0,00	Free to use
Space		
Electricity	400,00	100€/month x 4 months (duration of project)
Furniture	200,00	Table + Chair
Internet	280,00	70€/month x 4 months (duration of project)
Transport	0,00	Work from home
Total GC (Cost computed Generically)	**1200,00**	
Total Cost (Total CPA + Total GC)	**13657,80**	
Contingency	2048,67	Contingency margin = 15%
Total DC (direct cost) + IC (indirect cost) + Contingency	**15706,47**	
Data Generation Delay (1 Week)	240,45	Cost: Research psychologist, 14 hours. Risk: 50%
Data Generation Delay (2 Week)	48,09	Cost: Research psychologist, 14 hours. Risk: 10%
Total incidentals (or unforseen costs)	**288,54**	
TOTAL	**15995,01**	

Table 5: Final version of the budget structure [Own creation]

3 Identification of Laws and Regulations

Understanding the laws and regulations that have an impact on the design and development of the chatbot is one of the most crucial components of the project thesis.

3.1 Academic Regulations for the Degree Final Project

The UPC has a documentation of the regulations for the Degree Final Project available online [14]. This documents defines and describes the characteristics of the final project. The document also explains all the process needed to do the project. This document is of course very important and must be followed to ensure the correct development of the project.

3.2 GDPR Privacy Policy

A first step toward granting EU people and residents more control over how their data are used in organizations is the EU General Data Protection Regulation (GDPR). No matter where they are in the globe, businesses must adhere to the GDPR if they handle the personal data of people who reside in the EU.

A key requirement for businesses subject to the GDPR is that they make transparent and easily accessible information about the personal data they are processing available to the public. A clear and thorough privacy policy will help one to achieve this.

A privacy notice is a public statement from a company outlining how it manages customer information and adheres to data protection laws. A GDPR privacy notice is a crucial tool for assisting customers and users in making informed choices regarding the data you gather and use.

According to the GDPR [15], organizations are required to give customers a privacy disclosure that is:

- In a clear, visible, understandable, and readily available format
- Written in a straightforward manner, especially for any information aimed exclusively towards children
- Delivered on schedule
- Provided free of charge

In the following sections, all the information that must be included in a privacy notice is explained.

3.2.1 Company's contact details

Article 13(1)(a) [16] of the GDPR requires providing to users with: **”the identity and the contact details of the controller and, where applicable, of the controller's representative”**. An individual who determines how and why personal data is handled is referred to as ”the controller” or a ”data controller.”

Article 13(1)(b) [16] of the GDPR also requires providing: **"the contact details of the data protection officer, where applicable"**. A data protection officer is required for some firms of a specific size or those that consistently handle sensitive personal data (DPO).

3.2.2 The Purposes and Legal Basis for Processing

Article 13 (1)(c) [16] of the GDPR requires providing information about: **"the purposes of the processing for which the personal data are intended as well as the legal basis for the processing"**. To put it another way, is not allowed to process personal data unless there is a purpose for doing so. Additionally, there must be a legal justification for every form of data processing is carried out.

The GDPR sets out six legal bases at Article 6.

A person's personal data may only be processed if at least one of the following conditions is met [16]:

- You have their consent.
- To carry out or enter into a contract with them, you must process their personal data.
- It's required by law that you handle their personal information.
- Failure to process their personal data could endanger their lives or the life of another person.
- Processing their personal data is something you're doing in the public interest.
- You have a legitimate interest in processing their personal data.

The app falls under the category **"You have a legitimate interest in processing their personal data"** [16] since it collects user data in order to identify potential cognitive distortions based on user input.

3.2.3 Sharing of user's personal data

Article 13 (1)(e) [16] requires to provide information about: **"the recipients or categories of recipients of the personal data, if any"**. In the app's case the data is never shared with third party companies.

3.2.4 Sharing of user's personal data to a third country

Article 13(1)(f) [16] of the GDPR requires providing information about: **"the fact that the controller intends to transfer personal data to a third country or international organization and the existence or absence of an adequacy decision by the commission"**. A "third country" refers to a country outside of the EU.

The list of nations with "sufficient" data protection rules is maintained by the European Commission. You must indicate if a country is on the list if you are sending data to a third country. In the app's case the data is never shared with third countries.

3.2.5 Period of time storage of user's personal data

Article 13(2)(a) [16] of the GDPR requires informing users: **"the period for which the personal data will be stored, or if that is not possible, the criteria used to determine that period"**. It's crucial to comply with the GDPR's prohibition on keeping personal data longer than necessary. In the case of the app, user data is never stored.

3.2.6 User's Rights

Chapter 3 of the GDPR [17] sets out the rights that people have over their data. The GDPR not only requires you to not only make it easier for your users to access these rights, **but also to inform them of those rights in your Privacy Policy**. Additionally, you must let the user know how to file a complaint with their local data protection authority.

3.3 The EU Regulatory Environment of Medical Device Software Development

The International Medical Device Regulators Forum (IMDRF) defines SaMD (Software as a Medical Device) as **"Software intended to be used for one or more medical purposes that perform these purposes without being part of a hardware medical device"**. Taking into account this definition, the chatbot falls into this category.

The relevant General Safety and Performance Requirements (GSPRs) will have to be complied with by all software that falls within the Medical Device category. MDSW lawful manufacturers must put out a dossier or technical document (TD) for their product in order to prove conformity with GSPR. The details and explanations of the documents' contents are provided in the section that follows.

Applicable GSPRs mainly refer to one of the following general fields:

- **Quality Management System (QMS) requirements**. MDSW developers must work following a QMS methdology.
- **Risk Management System (RMS) requirements**. The basic objective of an RMS is to make sure that any potential dangers are recognized, categorised, and minimized without negatively influencing the device's risk-benefit ratio. To do this, the manufacturer must develop and implement a risk management strategy that accurately identifies all hazards related to the devices, establishes the necessary risk mitigation measures, and evaluates the effectiveness of each strategy.
- **Clinical Evaluation and Post-market surveillance requirements**. The clinical evaluation of a MDSW must be carried out following *MDCG 2020-1, guidance on Clinical Evaluation of MDSW*, and a Clinical Evaluation Report drafted providing the following information:
  - A valid clinical association of the software with the targeted clinical condition or physiological state, usually by means of literature references.
  - An analytical evaluation of the software to show that it is capable of processing data appropriately.

  - The software's output is then validated clinically to guarantee that it is accurate and dependable in the context of the clinical setting.

- **Usability requirements**. SW developers must make sure that as many user errors as possible are prevented via the user interface. *IEC 62366 Medical devices — Part 1: Application of usability engineering to medical devices* must be followed when planning and conducting usability tests for this. From a cybersecurity and safety standpoint, each and every one of the discovered user errors must be taken into account in the risk analysis and contributed to the risk management strategy and report. As with any other risk, preventive steps must be taken if the possibility of user errors cannot be entirely removed. Some of them include increasing training or adding particular warnings to the user handbook.

In addition, theses requirements are specifically applicable to MDSW:

- **Software lifecycle requirements**. The software lifecyle is described in *IEC 62304 Medical device software — Software life cycle processes*. It provides a series of steps that should be taken by SW developers. In Figure 13 we can see the Software Lifecycle.

- **Cybersecurity requirements**. This mainly regard patient data protection and protection from other cyber threats.

As with any other MD, it is advised that MDSW developers use approved techniques and standardized procedures like the ones listed below in order to adhere to the relevant GSPRs:

- **International standards** (mainly ISO [5], IEC [6] and ANSI/AAMI [7] standards)

- **MDCG** or **IMDRF** guidance documents

Table 6 provides an exhaustive list of the standards and guidelines now in existence that are advised to be followed by MDSW developers in order to accomplish with applicable GSPRs, together with their most recent updates.

[5] International Organization for Standardization (ISO)

[6] International Electrotechnical Commission (IEC)

[7] American National Standards Institute (ANSI)/ Association for the Advancement of Medical Instrumentation (AAMI)

4 Sustainability report

It is well known about climate change and the consequences that we are going to suffer or that we are even suffering now. Thus is really important for humans and companies to stop being selfish and think about the future of the world and cooperate in order to reduce pollution urgently. Thus, it is important to check the footprint of a project to see how does it impact in the environment. Assessing for the economical impact is also important, it helps us to optimize cost and savings. Finally keeping track of the social impact of a company/project is also very important. New technologies in particular have changed millions of people life including minorities and in developing countries.

4.1 Self assessment

Students were asked to complete a survey for their bachelor thesis. This survey asks respondents about their knowledge of sustainability in various fields. Some of these fields include economic, environmental, and social sustainability. After doing the survey, I realize that the Environmental field is my weak spot. In particular, I don't know which indicators to use to measure the impact in this aspect. Regarding the economical field, I have some intuition on how to measure and control the economical impact since I previously did a a budget. Finally in the social field, I think I know how new technologies and in particular my project impacts society. So in conclusion, I have a below average level of knowledge about sustainability, especially in the environmental field.

4.2 Environmental dimension

Regardind the PPP, As of 2023-01-10 19:00 according to Nowtricity a website that offers real time information of the emissions of every country [18], the current emissions in Spain is 137 grams C02 / kWh. The computer used for the project consumes an average of 0.2 kWh. Taking into account the previous information, the environmental impact can be calculated as shown in Formula 3.

$$137\frac{\text{gr CO2}}{kWh} \cdot 0.2 \text{ kWh} \cdot 540 \text{ h (total duration of the project)} = 14796 \text{ gr of CO2)} \tag{3}$$

One approach to reduce the the impact would be to execute in parallel to reduce training and evaluation time. This can be accomplished by increasing the njobs parameters above 0: the sklearn python library includes a parameter for determining the number of jobs to run in parallel for cross-validation.

Regarding the exploitation, as mentioned before, most people go to therapy in the traditional way. Since people can have access to KAI without the need to move from home, the project helps in reducing the pollution because they no longer need to take a mains of transport to go to a session. So we can conclude that KAI is more environmentally-friendly.

Regarding the risks, the project doesn't pose any, in fact the projects helps in reducing people's ecological footprint as mentioned previously.

4.3 Economic dimension

Regarding PPP, in section 2.3 the estimated costs of the project are identified and calculated and the budget is also shown. The hours required for the project have been revised and finally reduced resulting in saving costs.

Regarding the exploitation, nowadays most people go to in-person therapy with a professional therapist which is really expensive: in Spain the average price for one session is 50€ and taking into account that on average a person needs between 8-20 sessions, the final costs amounts to 400-1000€ . Online therapy sessions are getting popular giving people the flexibility to receive support and help with the need to transport. This option is usually more economical than traditional therapy (in person). Self-help therapy chatbots currently available in the market work on a free basis but if you want access to more content you need to subscribe, furthermore if you want to have access to a therapist: rates vary depending on the therapist or works on a subscription basis. Since KAI is free, it will help people embark in their self-help journey in therapy with the guide of the chatbot in a more affordable way.

In the future, the project will have an almost inevitable cost: human resources. This cost could be reduce with the automation of tasks and with the availability of datasets.

Regarding the risks, the project is very dependant on data, if the quality of data is not good enough it could lead to very inaccurate predictions.

4.4 Social dimension

Regarding the PPP, it has aided me in learning more about psychology, a field in which I have always been interested, and how technology can help people with their psychological needs. Furthermore, it has helped me in becoming more familiar with the machine learning field, particularly in the healthcare/medical sector, in which I am very interested. It also made me realize that Python is a powerful programming language, particularly for AI, due to the existence of an extensive library. What's more, the project experience has assisted me in determining whether I truly want to pursue a career in AI in healthcare or medicine. Finally, the most significant and special contribution of the project to me has been the opportunity to provide a mental health support tool like KAI to my sister, who unfortunately suffers from a severe mental illness.

Regarding the exploitation, the project will help to close the gap between those who need and those who receive mental health care. They will also have 24/7 access to support. More importantly, **it will allow people to have access to mental health support in a more affordable, autonomous, and time-efficient way**. This project is aimed at people who have mild to low symptoms of depression or anxiety. People who suffer from severe mental health or struggle with severe depression and anxiety are advised against using the chatbot.

Regarding the risks, as mentioned before, the project is not aimed at people who suffer from severe mental issues, instead is a tool for people with mild symptoms of anxiety and/or depression to do a self-guided CBT therapy.

5 Technical Competences

During the development of the project thesis, the following technical competences from computing specialization were followed:

CCO2.1
To demonstrate knowledge about the fundamentals, paradigms and the own techniques of intelligent systems, and analyse, design and build computer systems, services and applications which use these techniques in any applicable field. [Quite]

During the project design and development of an application (a chatbot) using machine learning has been done.

CCO2.2
Capacity to acquire, obtain, formalize and represent human knowledge in a computable way to solve problems through a computer system in any applicable field, in particular in the fields related to computation, perception and operation in intelligent environments. [Quite]

This competence was achieved with the acquisition of human knowledge and the representation through machine learning to detect possible cognitive distortions.

CCO2.3
To develop and evaluate interactive systems and systems that show complex information, and its application to solve person-computer interaction problems. [A little]

This competence was achieved with the design and development of the chatbot that extracts and shows complex information (possible cognitive distortions) from user input, a form of human-computer interaction (conversational user interface).

CCO2.4
To demonstrate knowledge and develop techniques about computational learning; to design and implement applications and system that use them, including these ones dedicated to the automatic extraction of information and knowledge from large data volumes. [In depth]

The study included extensive research on the top machine learning methods for topic classification. In addition, the design and development of the chatbot that automatically detects potential cognitive distortions using a machine learning technique.

6 Dialogue Flow

The design of the chatbot's dialogue flow is described in this section. The dialogue flow is divided in three main parts as shown in Figure 12 which are the following: **identifying potential cognitive distortions, challenging potential cognitive distortions, and chatbot's feedback**. The design of the dialogue flow is based on Module 9 and Module 10 of A Therapist's Guide to Brief Cognitive Behavioral Therpy [1].

6.1 Identifying Potential Cognitive Distortions

Finding cognitive distortions is the initial stage in the therapy's cognitive component. To accomplish this, one must first identify the user's automatic thoughts. An automatic thought is a thought that comes to you without conscious thought. Automatic thoughts are typically associated with negative emotions and might be triggered by certain events or circumstances.

The red elements in Figure 12 are a part of the first step, which is to identify potential cognitive distortions. The steps to accomplishing this are as follows:

- Asking how the user has been feeling. If the user's answer is positive then the support to therapy ends and the dialogue starts over. Otherwise the dialogue flow continues.
- Asking the user what made them feel this negative feeling.
- Asking the user what where they thinking when they were in that situation that made them feel bad in order to detect an automatic thought. In this part is where the ml model to detect potential cognitive distortions will be used.
- Asking the user how would they rate their negative feeling/mood that they previously mention in the first step.

6.2 Challenging Potential Cognitive Distortions

After identifying the potential cognitive distortions, the next step is challenging them. This will be done by using Dysfunctional Thought Record (DTR).

In cognitive-behavioral therapy (CBT), a Dysfunctional Thought Record (DTR) is a tool used to identify and address negative ideas and beliefs that contribute to emotional and behavioral problems. The DTR is a systematic form that assists people in recognizing their negative behaviors and beliefs, weighing the evidence supporting and refuting them, and coming up with more reasonable alternatives.

Figure 12 shows that the second step—challenging potential cognitive distortions—includes the green components. This is done by asking the the user for evidence that their thought is true and not true and finally asking the user to think of an alternative way to see the situation

The last step consist in asking the user to rate their mood after providing their alternative and more objective way of seeing the situation to see if there has been an improvement and the user feels better.

6.3 Chatbot's Feedback

The chatbot provides feedback to the user as the final stage of the conversation flow. This is accomplished by providing the user with a definition of the potential cognitive distortion that has been identified during the conversation, along with some helpful advice. If the chatbot notices that the user's mood hasn't improved, there is also a relaxing exercise (a link to a video).

6.3.1 Relaxation Exercise

Deep breathing is used in this part of the session. Slowing down the shallow, irregular breathing that usually occurs when people are agitated, worried, or anxious is the aim of deep breathing. The patient may have symptoms like hyperventilation and dizziness as a result of rapid and shallow breathing, which can cause their blood oxygen levels to fall and affect their ability to concentrate. A deep, complete breath may instead increase the amount of oxygen-rich blood flow, which may result in a sense of calm or slowness.

7 Dataset

The dataset of the project recollects phrases examples of 15 cognitive distortions (see Figure 11 to see the lists of cognitive distortions with their definitions). The dataset is composed of 595 rows and 2 columns. In the first columns there are example phrases of cognitive distortions and the second column indicates the type of cognitive distortion. Since not all automatic thoughts are negative or cognitive distortions, non-cognitive distortions examples have been added too.

As mentioned in section 2.1.3, the data has been generated by recollecting examples of cognitive distortions from trustful sources (books, articles, official psychology pages, etc).

7.1 Preprocessing

In this section the preprocessing done to the dataset is explained.

7.1.1 CountVectorizer and TfidfTransformer

Since machine learning models can't work with texts, it necessary to transform it into numerical representation. This is done by using two functions: CountVectorizer and TfidfTransformer. According to the official scikit library [19], CountVectorizer is used to **"Convert a collection of text documents to a matrix of token counts"**, while TfidfTransformer is used to **"Transform a count matrix to a normalized tf or tf-idf representation"**.

TF-IDF (Term Frequency-Inverse Document Frequency) is used to determine the importance of a word in a document or group of documents. By employing TF-IDF, it is intended to provide less weight to terms that are frequently used and more weight to words that are uncommon or unique to the documents under consideration.

The number of times a word appears in a document, normalized by the number of words in the document, is known as its frequency (TF). The logarithm of the total number of documents in the corpus divided by the number of documents where the word appears gives the inverse document frequency (IDF). By multiplying a word's TF and IDF values, one can determine the overall weight of the word in a document.

7.1.2 Balancing

When there are significantly less samples from one or more classes than there are from other classes, the dataset is said to be unbalanced. This might happen if the data were gathered from a real-world situation where there might not be an equal distribution of examples among the various classes. The dataset is clearly unbalanced, as can be seen in Figure 14 (there are almost 50 examples of Emotional Reasoning but only a little more of 20 examples of Always Being Right).

Machine learning algorithms may encounter difficulties when given unbalanced datasets because they may be biased in favor of the dominant class and may not adequately represent the minority class. This may result in models that are not generalizable to real-world scenarios and poor performance on the minority class.

The SMOTE methodology is used for balancing in this project. SMOTE (Synthetic Minority Oversampling Technique) is an oversampling technique used in machine learning to overcome the problem of unbalanced datasets. Instead of just duplicating existing examples, it generates synthetic examples of the minority class to balance the distribution of classes.

To generate synthetic examples, SMOTE first selects a minority class example and finds its K nearest minority class neighbors. It then interpolates a new synthetic example between the selected example and one of its neighbors, by sampling from a line between the two examples. This process is repeated until the desired amount of oversampling is achieved.

8 Model Experimentation

Each machine learning model that was used in the model experimentation is introduced in the subsections that follow, along with an explanation of how each hyperparameter was tuned. Additionally, the model experimentation results are presented and analyzed.

8.1 Multinomial Naive Bayes

The Multinomial Naive Bayes algorithm is a classification method that is based on the Naive Bayes algorithm and is specifically designed for text classification tasks with multiple classes. It estimates the probability of each class label occurring and the probabily of each feature occurring given a specific class label, and uses the Bayes theorem seen in Formula 4 to classify new data points.

The Bayes theorem can be expressed as follows:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{4}$$

where:

- $P(A|B)$ is the probability of event A occurring given that condition B is true.
- $P(B|A)$ is the probability of condition B being true given that event A has occurred.
- P(A) is the probability of event A occurring.
- P(B) is the probability of condition B being true.

8.1.1 *alpha* Hyperparameter

The alpha parameter is an Additive (Laplace/Lidstone) smoothing parameter. Laplace smoothing, often known as add-k smoothing or additive smoothing, is a method for keeping probability estimates from having zero probabilities. It is frequently used to enhance the performance of algorithms that rely on probabilistic estimations in natural language processing and machine learning applications, such as text classification and language modeling.

In probability estimation, zero probabilities can occur when a feature has not been observed in the training data. For example, in a text classification task, a word may not appear in the training data for a particular class label, resulting in a zero probability estimate for that word given the class label. This can cause problems when classifying new data points because the zero probability can result in a zero probability for the entire data point, regardless of the other features.

Laplace smoothing solves this problem by increasing the count of each feature by a small constant called the alpha parameter, also known as the ”smoothing parameter” or ”smoothing factor”. As a result, there is a slight rise in the probability estimates for all attributes, even those with zero probabilities.

We must specify a value for the smoothing parameter, which controls how much smoothing is applied to the probability estimates, in order to perform Laplace smoothing. The smoothing option is frequently set to 1, which increases the count of each feature by one. This can be mathematically expressed with Formula 5 where:

- $P(feature|class)$ is the probability of the feature occurring given the class label.
- count(feature, class) is the number of times the feature has been observed in the training data for the class label.
- count(class) is the total number of observations for the class label.
- alpha is the smoothing parameter.
- num_features is the total number of unique features in the training data.

$$P(feature|class) = \frac{count(feature, class) + \alpha}{count(class) + \alpha \cdot num\_features} \tag{5}$$

Laplace smoothing can help probabilistic models function better by lessening the effect of zero probability and preventing overfitting to the training set of data. To prevent adding too much bias to the probability estimates, it is crucial to select an acceptable value for the smoothing parameter.

8.2 Multinomial Logistic Regression

The Multinomial logistic regression algorithm is one of the models used in the experimentation for the automatic detection of potential cognitive distortions. Multinomial logistic regression is a classification method that is used to predict a categorical dependent variable, with multiple categories, from one or more independent variables.

It is assumed that the dependent variable in multinomial logistic regression has multiple categories, each of which may be predicted based on the values of the independent variables. The category with the highest probability is selected as the predicted outcome according to the model's estimates of each category's likelihood.

The model is based on the assumption that the log-odds[8] of the dependent variable are a linear combination of the independent variables. This is expressed in Formula 6 where y is the dependent variable, k is a category of y, $x_1, x_2, ..., x_n$ are the independent variables, and $b_0, b_1, b_2, ..., b_n$ are the coefficients that are estimated by the model.

$$\log\left(\frac{p(y=k)}{1-p(y=k)}\right) = b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + ... + b_n \cdot x_n \tag{6}$$

Maximum likelihood estimation is the most used technique for calculating the beta parameter, or coefficient, in this model (MLE). This method tests several beta values repeatedly in search of the best match for the log odds. Logistic regression aims to maximize this function after each of these iterations in order to determine the optimal parameter estimate.

[8] The probability of success divided by the probability of failure

Once the optimal coefficient (or coefficients, if there are numerous independent variables) has been identified, the conditional probabilities for each observation can be computed, logged, and summed to obtain a predicted probability.

There are several advantages to using multinomial logistic regression, including its ability to handle multiple categories and its ability to model the relationships between the independent variables and the dependent variable. It's crucial to keep in mind that the model makes the assumption that the independent variables are unrelated to one another, which may not always hold true in actual life.

8.2.1 *penalty* Hyperparameter

According to the scikit-learn API library [13] the penalty parameter is used to **"Specify the norm of the penalty"**.

Regularization is a method for avoiding overfitting in machine learning models like multinomial logistic regression. Regularization is used to impose a **penalty** on the model's complexity, which helps to lower the variance and enhance the model's generalization capabilities. Regularization is accomplished in multinomial logistic regression by including a penalty term in the objective function that is being optimized.

8.2.2 *C* Hyperparameter

According to the scikit-learn API library [13] the C hyperparameter is the **"Inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization"**.

8.2.3 *solver* Hyperparameter

The solver hyperparameter in multinomial logistic regression is a parameter that determines the algorithm used to optimize the model.

8.3 Support Vector Machine

A Support Vector Machine (SVM) is a type of supervised learning algorithm that can be used for classification or regression tasks. The algorithm finds the best boundary (a hyperplane) that separates the data into different classes. The boundary is chosen in a way that maximizes the margin, which is the distance between the boundary and the closest data points from each class (these points are called support vectors). The goal of SVM is to identify a boundary that effectively divides the classes while also having the biggest margin to reduce generalization error as shown in Figure 15.

Hyperplanes act as judgment lines for categorizing the data points. Different classes can be given to the data points that fall on each side of the hyperplane. Additionally, the number of features affects the hyperplane's dimension. For instance, if there are only two input features, the hyperplane is only a line, and if there are three input features, the hyperplane becomes a two-dimensional plane.

The hyperplane is defined by a weight vector (w) and a bias term (b). The equation of the hyperplane is given by Formula 7 where x is a feature vector and w and b are the parameters of the hyperplane.

$$w \cdot x + b = 0 \tag{7}$$

The distance of a point x from the hyperplane is given by Formula 8 where $||w||$ is the norm of the weight vector.

$$distance = \frac{w \cdot x + b}{||w||} \tag{8}$$

An SVM's objective is to determine the hyperplane with the greatest margin—that is, the distance between the hyperplane and the nearest data points from either class—and the hyperplane that maximum separates the classes.

The SVM algorithm uses a method known as the ”kernel trick” to find the hyperplane. The input data is mapped into a higher-dimensional space using the kernel method, making it simpler to locate the hyperplane. The type of data and problem complexity determine the kernel function that is employed. The linear, polynomial, and radial basis functions are often employed kernel functions.

After locating the hyperplane, the SVM can be used to categorize additional data points by determining how far they are from the hyperplane. The point is categorized as belonging to one class if the distance is positive, and to the other class if the distance is negative.

SVMs have several advantages over other classification algorithms. They are robust to noise and can handle high-dimensional data. They have also been extensively investigated and employed in a wide range of applications. They also have a strong mathematical foundation.

8.3.1 *decision_function_shape* Hyperparameter

Support Vector Machines (SVMs) employ the decision function to categorize data points according to their separation from the hyperplane. Based on the trained SVM model, predictions about the class labels of new data points are made using the decision function.

The decision function is defined with Formula 9 where w is the weight vector, x is the feature vector of the data point, and b is the bias term. The sign of the decision function value determines the class label assigned to the data point. If the decision function value is positive, the data point is assigned to one class, and if it is negative, it is assigned to the other class.

$$f(x) = w \cdot x + b \tag{9}$$

8.4 K-Nearest Neighbour

K-Nearest Neighbors (KNN), one of the most widely used machine learning algorithms, is a simple and easy to implement machine learning algorithm for classification and regression issues.

In the KNN algorithm, a new data point is classified based on the majority class of its "nearest neighbors". The number of neighbors, "K", is a hyperparameter that is specified by the user.

To classify a new data point, the KNN algorithm follows these steps:

- Calculate the distance between the new data point and all the training data points.
- Select the K training data points that are closest to the new data point.
- Determine the majority class of the K nearest neighbors.
- Assign the new data point to the majority class.

8.4.1 *n_neighbors* Hyperparameter

According to the scikit-learn API library [13] the n_neighbors hyperparameter specifies the **"Number of neighbors to use by default for kneighbors queries"**.

8.4.2 *weights* Hyperparameter

In the K-Nearest Neighbors (KNN) algorithm, the weights of the nearest neighbors can be used to give more or less influence to certain data points when making a prediction.

8.4.3 *p* Hyperparameter

According to the scikit-learn API library [13] the p hyperparameter is the **"Power parameter for the Minkowski metric. When p = 1, this is equivalent to using manhattan_distance (l1), and euclidean_distance (l2) for p = 2. For arbitrary p, minkowski_distance (lp) is used"**.

8.5 Random Forest

Random forest is a machine learning algorithm that combines the output of multiple decision trees to reach a single result. The method combines both bias and variance reduction techniques by constructing a large number of decision trees and then aggregating their predictions.

Decision trees are a common supervised learning method used in regression and classification problems [19]. They are known as decision trees because they are constructed using a tree-like structure, with leaf nodes serving as the output or prediction and inside nodes indicating decisions based on the value of input features.

In decision trees, impurity refers to how mixed or "impure" the data is with regard to the target labels in a given node or subset of the data. When developing a decision tree, the objective is to develop a model that can precisely predict the target label for a given input. Typically, this is done by dividing the data into subsets with as pure a target label distribution as is possible.

Decision trees frequently employ the Gini impurity, entropy, and misclassification rate among other impurity measures.

Gini impurity is a measure of the probability of misclassifying a randomly chosen element in a set, and is defined with the Formula 10 where p(i) is the proportion of elements in the set that belong to class i and n are the number of elements in the set.

$$\text{Gini impurity} = 1 - \sum_{i=1}^{n} p(i)^2 \tag{10}$$

Entropy is a measure of the amount of uncertainty in a set, and is defined with the Formula 11 where p(i) is the proportion of elements in the set that belong to class i and n are the number of elements in the set.

$$\text{Entropy} = -\sum_{i=1}^{n} p(i) \cdot log(p(i)) \tag{11}$$

Misclassification rate is simply the number of misclassified elements in a set divided by the total number of elements in the set.

8.6 Model Performance

In this section the model performance is presented and evaluated.

8.6.1 Hyperparameters Tuning

In order to avoid or at least reduce the impact of bias, Grid Search and Cross-Validation is used for the hyperparameters tuning. Grid search is a method for hyperparameter optimization that involves training and evaluating a model using a combination of different hyperparameter values, and selecting the combination that provides the best performance. Cross-validation is a method for evaluating the performance of a model by dividing the data into training and validation sets, training the model on the training set, and evaluating the model on the validation set. The data is split into k-folds in the k-fold Cross Validation form, and the model is trained on k-1 of the folds before being tested on the last fold. The test set is a new fold each time, and this operation is done k times. To estimate the model performance, the results are then averaged.

It's important to note that in every model, the k-neighbors hyperparameters from SMOTE (see Section 7.1.2 for a detailed explanation) is additionally tuned. This parameter controls the number of nearest neighbors used to generate synthetic samples. Increasing k_neighbors will make the synthetic samples more similar to the original samples, while decreasing k_neighbors will make the synthetic samples less similar to the original samples. **The different values of the parameter to be tuned are the following: [5,6,7,8,9,10]**.

In Figure 6 we can see the results obtained from the Hyperparameters Tuning of the different models. The Multinomial Naive Bayes has a low alpha as a best hyperparameter value which means that the model is less smooth, in other words, the model is less likely to assign a non-zero probability for an unseen feature in the training data. For the Multinomial Logistic Regression has a low C as best hyperparameter value. This low value indicates that there is less regularization which means that the model is more likely to fit the training

[9]. On the other hand, for the Support Vector Machine a very high C value was the best value, meaning there is more regularization. Finally, for the KNN (K-Nearest Neighbor), the euclidean distance was the best hyperparameter (p hyperparameter = 2).

Model	Best SMOTE Hyperparameter	Hyperparameters	Best Hyperparameters	Best Accuracy Through GridSearchCV
Multinomial Naive Bayes	k_neighbors: 10	alpha: [0.01, 0.1, 0.3, 0.5, 1.0]	alpha: 0.1	0.68
Multinomial Logistic Regression	k_neighbors: 2	penalty: ['l1', 'l2', 'elasticnet', 'none']	penalty': 'none'	0.69
		C: [0.01, 0.1, 0.3, 0.5, 1.0,]	C: 0.01	
		solver: ['lbfgs', 'newton-cg', 'sag', 'saga']	solver: 'saga'	
Support Vector Machine	k_neighbors: 10	decision_function_shape: ['ovo', 'ovr']	decision_function_shape: 'ovo'	0.64
		C: [0.01, 0.1, 0.3, 0.5, 1.0,]	C: 1.0	
Random Forests	k_neighbors: 3	-	-	-
K-Nearest Neighbour	k_neighbors: 10	n_neighbors: [5,6,7,8,9,10,]	n_neighbors: 5	0.54
		weights: ['uniform', 'distance']	weights: 'distance'	
		p: [1, 2, 3]	p: 2	

Figure 6: Results of the Hyperparameters Tuning [Own Creation].

8.6.2 Evaluation

For evaluating the performance of the different models, the analysis will be done between four metrics: Accuracy, Precision, Recall, F1-Score. Note that the results to be analyzed are the weighted average of the metric mentioned before to take into account the number of instances of each class in the testing. The weighted average is specially preferred when there is an imbalance in the testing data.

Accuracy is a more simple metric that doesn't take into account the cost of having a missclassification. Accuracy is expressed with Formula 12.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{12}$$

The precision measures how many positive predictions are actually positive (True positive). This metric is particularly appropriate when the cost of False Positive is high. For instance, in the case of spam detection, a false positive indicates that an email that's it's actually important (no spam) has been classified as spam and thus the user is losing valuable information. Precision is expressed with Formula 13.

$$Precision = \frac{\text{Number of True Positive}}{\text{Total predicted positive}} \tag{13}$$

The recall measures how many of the actual positives are predicted as positive. This metric is the most appropriate when there is a high cost of False Negative. For example, if a person with cancer is labelled as not having cancer, the consequences could be disastrous. Recall is expressed with Formula 14.

$$Recall = \frac{\text{Number of True Positive}}{\text{Total actual positive}} \tag{14}$$

The F1-Score gives us a balance between Recall and Precision and is expressed with Formula 15.

[9]Fitting the training data refers to the process of adjusting a model's parameters so that it accurately predicts the output values for the input values in the training set.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

As we can see in Figure 7, all the models had a better performance in the precision metric. In other words, they have a lower False Positive rate. Models with high precision can be considered ”conservative” since it will only predict a positive sample when it’s very confident that is actually positive. The Support Vector Machine is particularly conservative since it has the highest precision with a 0.75 score.

Furthermore, all of the models scored lower in the F1-Score. This could be an indication that the models struggle in correctly identifying positive examples, in other words, there is a low recall. This can also be caused when the class distribution is imbalanced (specially in the training set).

Overall, the KNN (K-Nearest Neighbour) had the worst performance. This bad result could be an indication that the dataset generated is complex [10] since KNN is better suited for small and more simple datasets.

On the other hand, both Random Forest and Multinomial Naive Bayes curiously had the same performance except for the GridSearchCV where Multinomial Naive Bayes had a slightly better score. Both models are overall the best models. The final model chosen to integrate to the implementation of the chatbot is Random Forest. The criteria to make the tiebreaker between the two models was done by taking into account the following:

- **Complexity**: Random forests are more complex than multinomial Naive Bayes, as they involve building and training multiple decision trees. On more complicated datasets, though, they can frequently attain higher accuracy.
- **Feature Importance**: The relevance of each feature in the model can be determined by random forests [11], which is helpful for feature selection or for comprehending the model’s decision-making process. This kind of information is not provided by Multinomial Naive Bayes. This is crucial for understanding how the model ”made the decision” or ”justified” of choosing an specific cognitive distortion.

MODEL	BEST PARAMETERS	ACCURACY	PRECISION	RECALL	F1-SCORE	GRIDSEARCHCV
Random Forest	-	0.69	0.71	0.69	0.68	0.67
Multinomial Naive Bayes	*alpha: 0.1*	0.69	0.71	0.69	0.69	0.68
Support Vector Machine	*C: 1.0, decision_function_shape: ovo*	0.67	0.75	0.67	0.67	0.64
Multinomial Logistic Regression	*C: 0.01, penalty: none, solver: saga*	0.65	0.65	0.65	0.64	0.69
K-Nearest Neigbhour	*p: 2, weights: distance, k_neighbors: 10*	0.53	0.55	0.53	0.51	0.54

Figure 7: Model Experimentation’s Results [Own Creation]

[10] A complex dataset is one that has a large number of examples, a large number of features, or a high degree of complexity or non-linearity in the relationships between the input variables and the output variable.

[11] The importance of each feature in a random forest model can be determined by examining the amount by which the model’s accuracy declines when the values of that feature are randomly permuted. This is known as permutation importance, and it provides a way to quantify how important each feature is to the predictions made by the model.

9 Implementation

In this section, the implementation of the project is going to be explained. The implementation has been divided between the model experimentation and the chatbot implementation (the interface and the dialogue flow). The implementation has been done entirely in python and Jupyter Notebooks. I have decided to use the Python programming language because it has an extensive libraries specially for machine learning, it's easy to learn and use and due to the simplicity of the syntax, it's also quicker to code.

9.1 Model Experimentation

For every model a pipeline [12] is used to preproces the data as explained in Section 7.1, using CountVectorizer and TfidfTransformer to transform text into numbers. This is necessary since the models can't understand text, only numbers, as explained before. Smote is also used to balance the data and finally the model is also added to the pipeline.

Furthermore, the *GridSeacrhCV* function is called to tune the hyperparameters previously chosen and evaluating the results with Cross Validation with 10 folds in order to avoid/reduce the bias.

9.2 Chatbot Implementation

The implementation of the chatbot follows the dialogue flow shown in Figure 12 where the user explains his situation, his thoughts and rates his mood. Then Kai uses the machine learning model to detect the potential cognitive distortions in the user thought's.

However, to make the dialogue flow more natural, a set of rules have been defined to detect the intentions of the user following the rule-based chatbot design model. This is done by provinding a database of responses and giving the chatbot a set of rules to decide on how to choose to response from the database.

The main two libraries used for the implementation are: Natural Language Toolkit (NLTK) and Regular Expression (RegEx). Natural Language Toolkit is a Python library to work with human language data and Regular Expression is a sequence of characters that specifies a search pattern in text (Python supports RegEx with the re library).

The interface has been implemented with TKinter a python framework. This framework has been chosen because its really easy to use and has compatibility with the backend since it's in python too.

9.2.1 Rule-Based Chatbot

Following the rule-based methodology, the chatbot will search for specific keywords in the user input during the conversation. The keywords are crucial to understand what does the user intent to do. Once the intent is identified, the chatbot simply matches the intent with the predefined response.

[12] A pipeline is a tool for building and evaluating machine learning models. It's a sequence of transforms and a final estimator. The transofrms are applied in sequence to the input data, and the final estimator is used to make a prediction.

The code has the following structure:

1. **Importing Libraries**: The first step is importing all the required libraries. The re library is the package that handles regular expressions as mentioned before. The wordnet from the NLTK library is also going to be used. Wordnet is a lexical English database that defines semantics relationships between words (useful to find the meaning of the words, synonyms, antonyms, etc ...). The main purpose of wordnet for the project is to build up a dictionary of keywords to the keywords. This will allow me to avoid manually expand having to introduce every possible alternative word a user could use to match the specific keyword. The itertools is a module that allows us to handle the iterators in an efficient way.

2. **Building the Keyword List**: This part consists of building the list of keywords that the chatbot needs to look for.

3. **Building a dictionary of Intent**s: Once we have the list of keywords, the next step is building a dictionary of intents to match intents with keywords.

4. **Defining a dictionary of responses**: Consists on giving a list of predefined responses for each intent.

5. **Matching Intents and Generating Responses**: In this part the user input is taken and evaluated to see if there is any keyword. This is done with the RegEx search function. If there is no keyword match, the dialogue flow is continued (see Figure 12).

9.3 Dialogue example

In this section the chatbot is tested. There are two tests, one testing that the dialogue flow works correctly and the other one tests if the chatbot correctly detects the keywords previously designed and responds accordingly.

9.3.1 Dialogue Flow Test

In order to test the chatbot, an script has been written. The script is based on an example given in *A Therapist's Guide to Brief Cognitive Behavioral* [1], a book. The example is about a person of advanced age who feels he has nothing to offer to their family because the person was out of breath while playing with their grandchild.

Identifying Potential Cognitive Distortions

As we can see in Figure 8, the conversation is started by Kai by greeting and asking how the user feels. After receiving the user's answer, the chatbot detects that the feeling is negative with the aid of the SentimentAnalyzer of the NLTK library. Then the Kai asks the user to explain more and what was the person thinking at the time. This last person is really important since is the automatic thought that is going to be used for the model to detect the potential cognitive distortion.

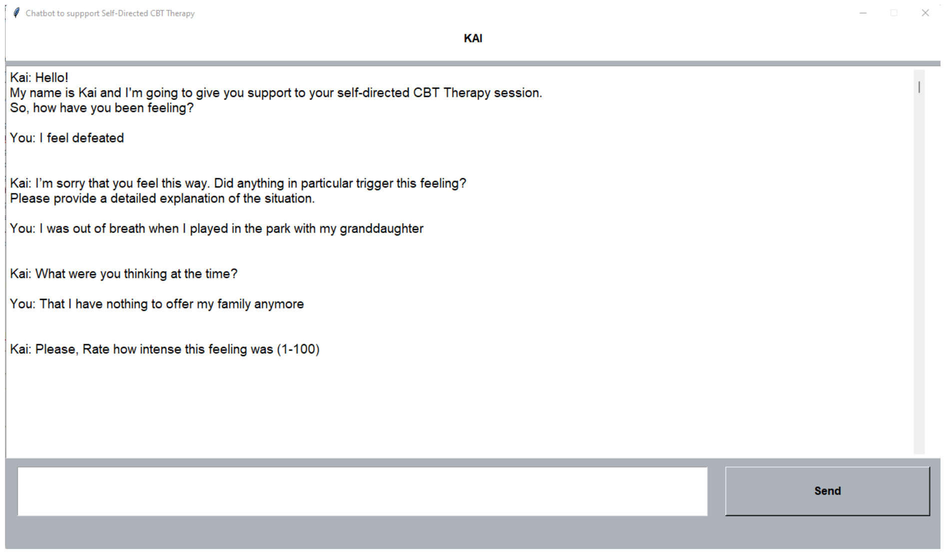

Figure 8: Screenshot showing the first part of the Dialogue flow [Own Creation]

Challenging Cognitive Distortions

As we can see in Figure 9, the second part of the dialogue starts which consists in challenging the Potential Cognitive Distortion by asking questions to make the user rationalize and think of an alternative way of seeing things. This is a fundamental part of the CBT therapy. Towards the end of the conversation, Kai tells the user the potential cognitive distortion detected by the model and gives some tips. If the mood is not improved, which means that the user doesn't feel any better after the session, a relaxation exercise is provided.

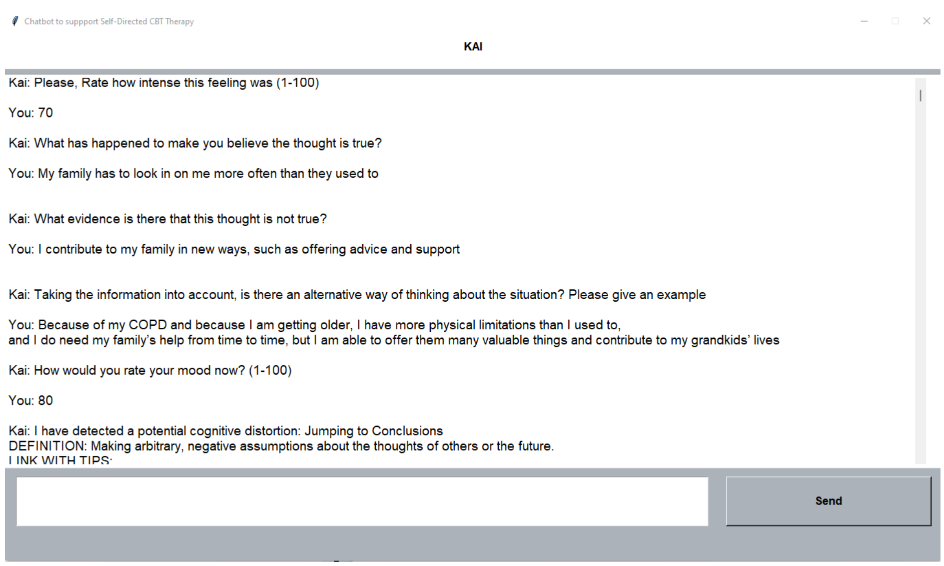

Figure 9: Screenshot showing the second part of the Dialogue flow [Own Creation]

9.3.2 Rule-Based Test

As seen in Figure 10, the chatbot correctly detects the keywords and responds accordingly.

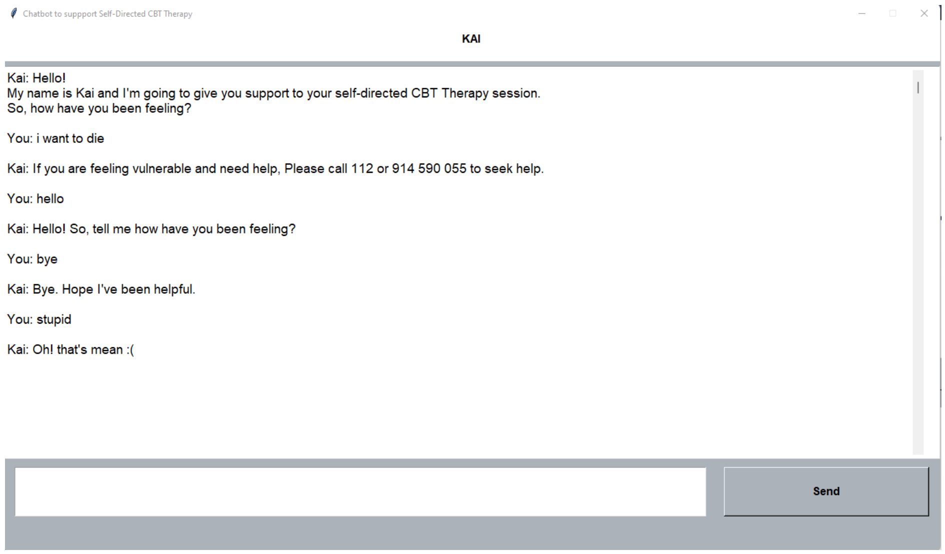

Figure 10: Screenshot showing how the chatbot correctly detects the the keywords and responds accordingly [Own Creation]

10 Conclusions

The main objective of the project was designing and developing a chatbot to support therapy for people who have mild to moderate sympotms of mental health issues. To achieve that a through research has been done for the design of the dialogue flow following the Cognitive Behavioral Therapy (CBT) methodology.

Finding an appropriate dataset for training the machine learning models for the automatic detection of potential cognitive distortions in text presented a significant challenge for the project. Unfortunately, there isn't a publicly accessible dataset, thus it was decided to generate one by gathering examples of cognitive distortions from reliable sources. The resulting dataset was small and could have been biased. SMOTE and cross validation were used to resolve this.

Furthermore, a review the different hyperparameters that could significantly impact the performance of the various models for model experimentation has been done for the hyperparameter tuning. We can conclude that the objective of detecting cognitive distortions was accomplished based on the performance that the model experimentation (see Figure 7) appears to have achieved.

Finally the chatbot development was done in python and the GUI with TKinter, a python's framework. Additionaly, to make the dialogue flow more dynamic, a set of rules have been defined, to make the chatbot respond accordingly when it detects that the user is doing certain actions, for instance greeting.

10.1 Future Work

Since the quality of the dataset affects the model's performance, it is essential to increase the dataset's size and quality by gathering more representative examples. This might be done with the use of crowdsourcing, by asking experts to provide examples, or by asking them to label phrases on open forums like Reddit.

Additionally, as the project developed, I came to understand that there are situations in which more than one cognitive bias could be present in a single sentence and that some cognitive biases are more connected to one another than others, making it harder to distinguish between them. To classify a sentence into more than one cognitive distortion, it would be interesting to perform a multilabel text classification. Additionally, it would be interesting to revise the different categories of cognitive distortions and establish a "super group" that aggregates all related cognitive distortions.

10.2 Reflexions

I was inspired to do this project because I thought it would be interesting to use AI in health care. After completing this project, I have come to the conclusion that the hype surrounding machine learning has blinded me. In reality, machine learning models are highly dependent on data, both in terms of quantity and quality.

It is particularly challenging to apply AI successfully in industries like healthcare, where data is limited and there are specific regulations with which we must comply. Another challenge is that most machine learning models end up being "black boxes" (it is difficult to understand how the model makes the decision based on data). Furthermore, the fact that we must train the models can be time-consuming, exceedingly error-prone, and frequently results in failure due to poor data quality. In fact, according to TechRepublic [20], about 85% of AI projects fail in bussines settings and most causes include bad data quality and problems with data labelling.

If we look at the definition of AI, is a discipline that strives or looks for designing computer systems that mimic human intelligence without the need of human intervention (except for the fact that you have obtain the data, clean it and then in some cases manually label it). While working on the project, I couldn't help but wonder whether babies and young children needed to see many dogs in order to recognize them. How do people identify things? Understanding how our brains truly function and how we learn things is perhaps the first step toward developing "real" AI. After all, using massive amounts of data to train the models may not be the best strategy; perhaps there is another way. Perhaps neuroscience holds the key to real AI development.

A 15 Major Cognitive Distortions

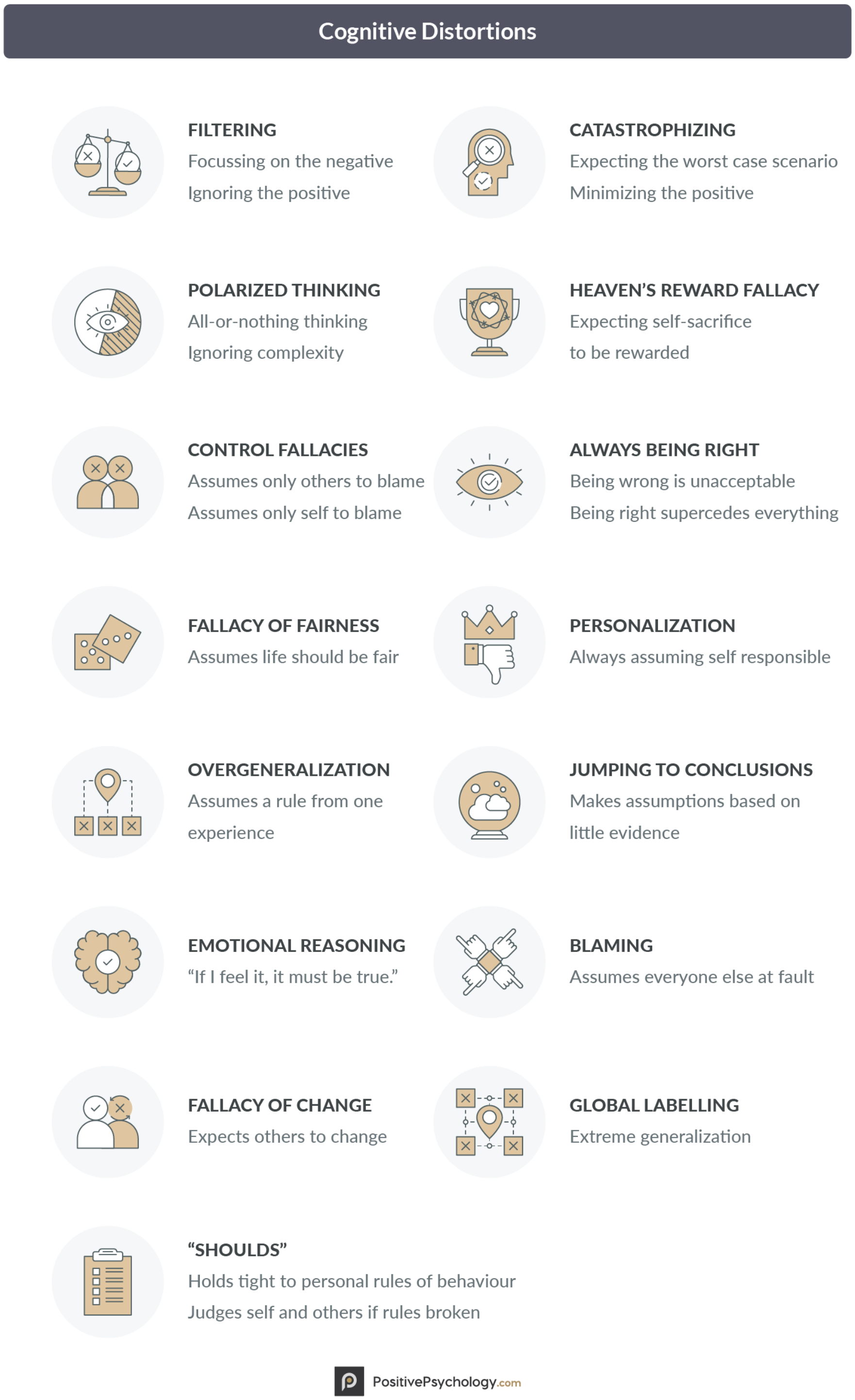

Figure 11: 15 major cognitive distortions by PositivePyschology, a website [4]

B Dialogue Flow

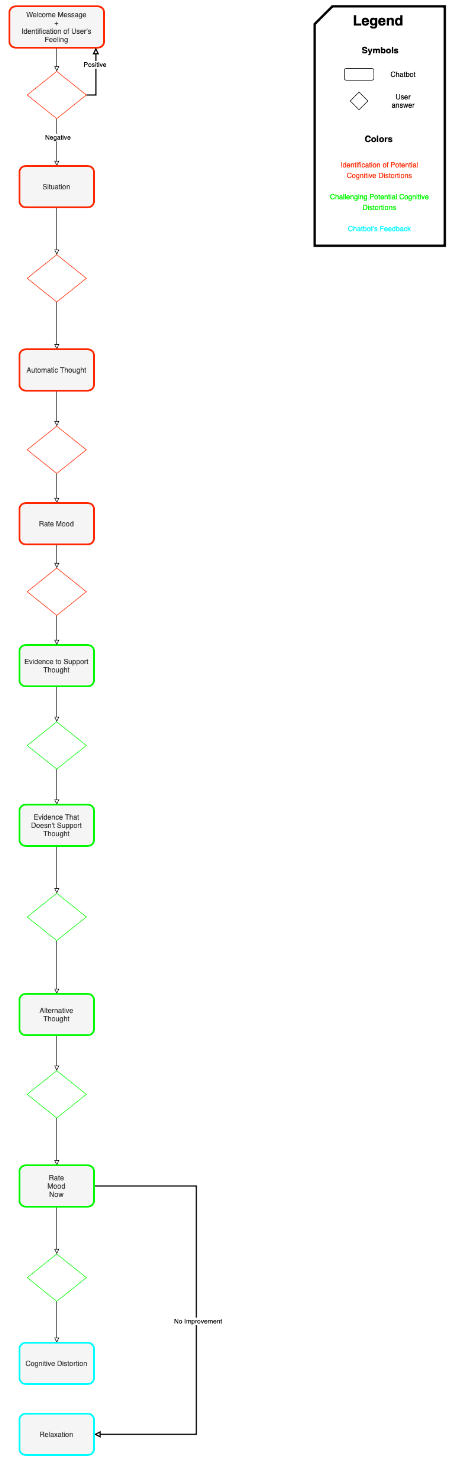

Figure 12: Chatbot's Dialogue Flow [Own Creation]

C EU Regulatory Environment for MDSW Development

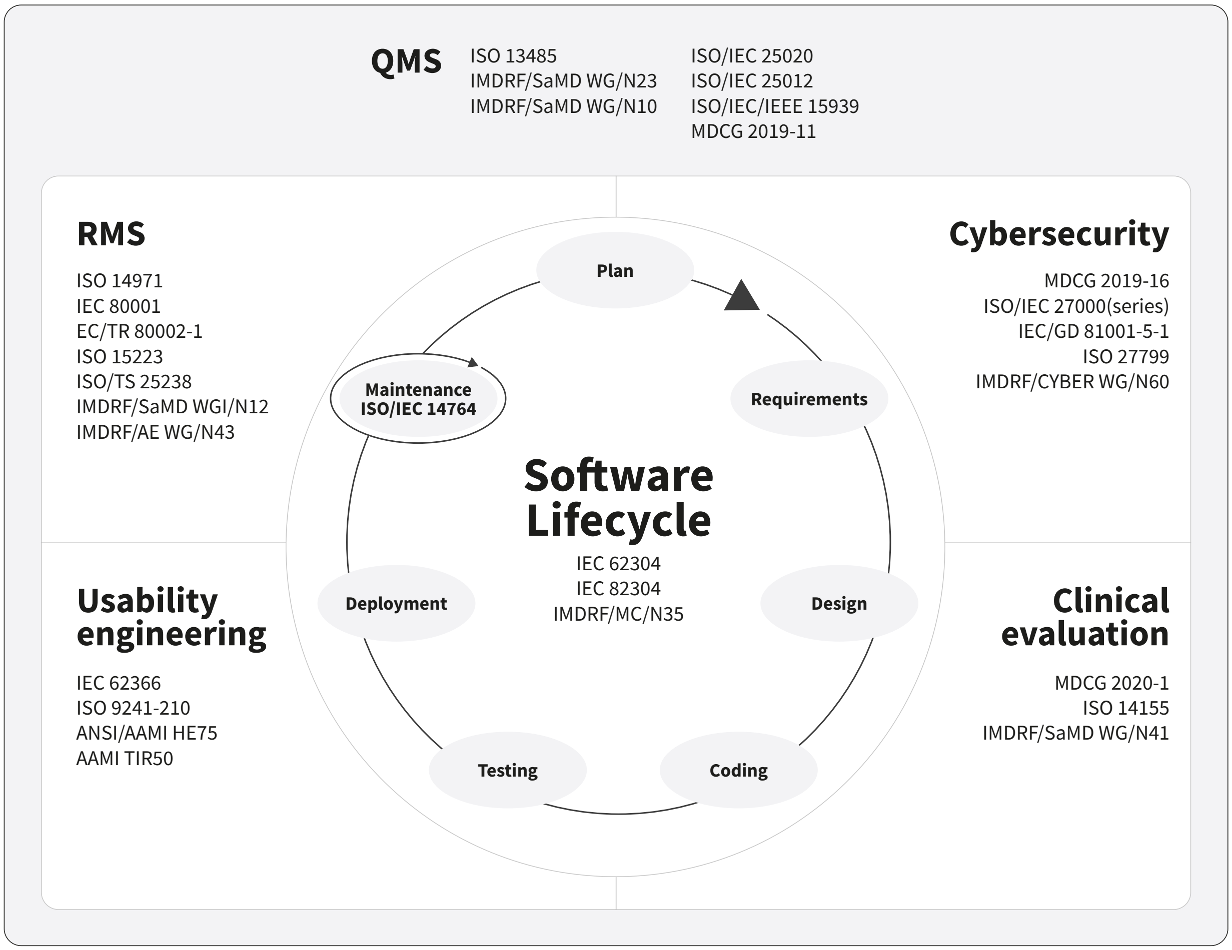

Figure 13: EU Regulatory Environment for MDSW Development. [5]

D Standard and guidance documents useful to demonstrate MDSW compliance with MDR

Requirement	Standard or Guidance	Title
Quality Management System (QMS)	**EN ISO 13485:2016/AC:2018 (*)**	Medical devices - Quality management systems - Requirements for regulatory purposes
	ISO/IEC 25020:2019	Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measurement framework
	ISO/IEC 25012:2008	Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Data quality model
	ISO/IEC/IEEE 15939:2017	Systems and software engineering — Measurement process
	MDCG 2019-11	Qualification and classification of software - Regulation (EU) 2017/745 and Regulation (EU) 2017/746
	IMDRF/SaMD WG/N23 FINAL: 2015	Software as a Medical Device (SaMD): Application of Quality Management System
	IMDRF/SaMD WG/N10 FINAL:2013	Software as a Medical Device (SaMD): Key Definitions
Risk Management System (RMS)	**EN ISO 14971:2019 (*)**	Medical devices - Application of risk management to medical devices
	IEC 80001-1:2010 (series)	Application of risk management for IT-networks incorporating medical devices — Part 1: Roles, responsibilities and activities
	EC/TR 80002-1:2009	Medical device software — Part 1: Guidance on the application of ISO 14971 to medical device software
	ISO/TS 25238:2007	Health informatics — Classification of safety risks from health software
	EN ISO 15223-1:2016 (*)	Medical devices - Symbols to be used with medical device labels, labelling and information to be supplied - Part 1: General requirements
	IMDRF/SaMD WG/N12 FINAL:2014	"Software as a Medical Device": Possible Framework for Risk Categorization and Corresponding Considerations
	IMDRF/AE WG/N43 FINAL:2020 & Annexes	IMDRF terminologies for categorized Adverse Event Reporting (AER): terms, terminology structure and codes
Clinical Evaluation	**EN ISO 14155:2020 (*)**	Clinical investigation of medical devices for human subjects - Good clinical practice
	MDCG 2020-1	Guidance on clinical evaluation (MDR) / Performance evaluation (IVDR) of medical device software
	IMDRF/SaMD WG/N41 FINAL:2017	Software as a Medical Device (SaMD): Clinical Evaluation
Cybersecurity	**ISO/IEC 27000:2018(en) (series)**	Information technology — Security techniques — Information security management systems — Overview and vocabulary
	ISO 27799:2016	Health informatics — Information security management in health using ISO/IEC 27002
	IEC/CD 81001-5-1 (draft 2021)	Health software and health IT systems safety, effectiveness and security — Part 5-1: Security — Activities in the product lifecycle
	MDCG 2019-16 rev.1	Guidance on cybersecurity for medical devices
	IMDRF/CYBER WG/N60FINAL:2020	Principles and Practices for Medical Device Cybersecurity
Usability	**IEC 62366-1:2015 (*)**	Medical devices - Application of usability engineering to medical devices
	ISO 9241-210:2010	Ergonomics of human-system interaction - Human-centered design for interactive systems
	ANSI/AAMI HE75:2009/(R)2018 (*)	Human factors engineering- Design of medical devices
	AAMI TIR50:2014 (*)	Post-market surveillance of use error management
Software lifecycle	**EN 62304:2006/AC:2008(*)**	Medical device software - Software life-cycle processes
	IEC 82304-1:2016	Health software — Part 1: General requirements for product safety
	ISO/IEC 14764:2006	Software Engineering — Software Life Cycle Processes — Maintenance
	IMDRF/MC/N35 FINAL:2015	Statement regarding Use of IEC 62304:2006 "Medical device software -- Software life cycle processes"
(*) Although they are not software-specific, these standards are highly relevant for the development of MDSW.		

Table 6: Standard and guidance documents useful to demonstrate MDSW compliance with MDR. [5]

E Number of examples grouped by cognitive distortions

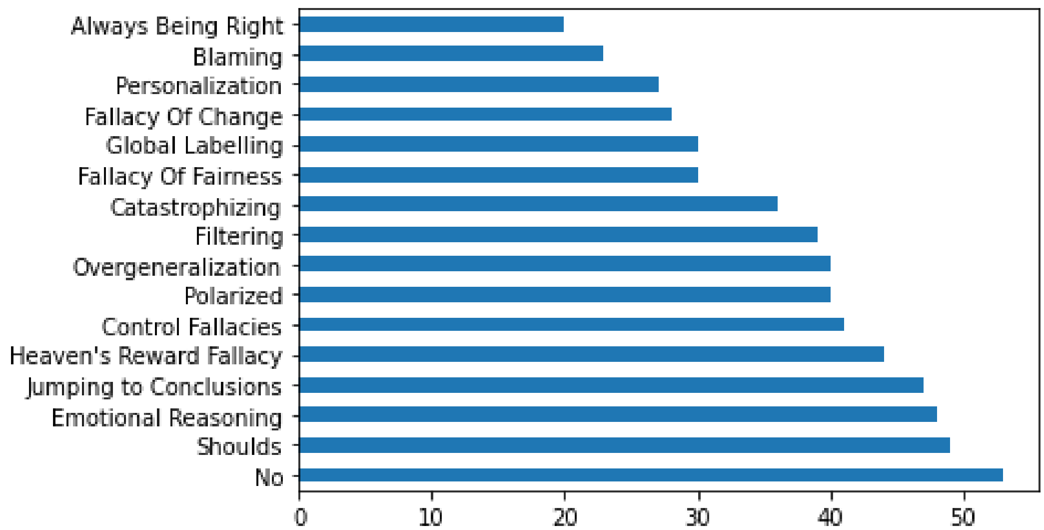

Figure 14: Number of examples grouped by cognitive distortions [Own Creation]

F Support Vector Machine Hyperplanes

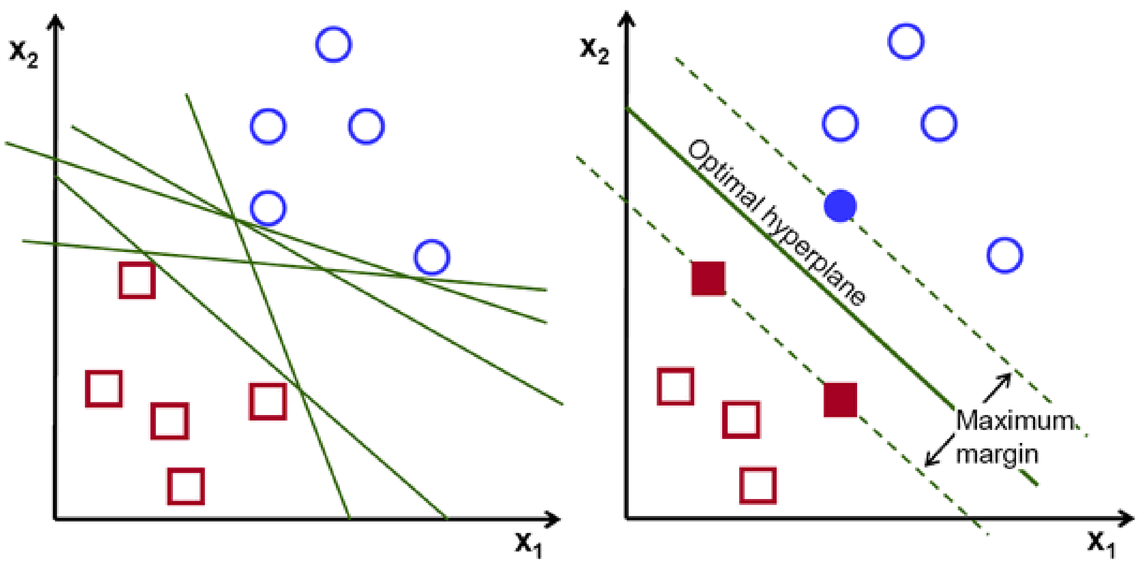

Figure 15: There are numerous alternative hyperplanes that might be used to divide the two groups of data points (left image). Finding a plane with the greatest margin—that is, the greatest separation between data points from both classes—is our goal (right image) in SVM. Images obtained from Towards Data Science [6], a website.

References

[1] J. A. Cully and A. L. Teten. *A Therapist's Guide to Brief Cognitive Behavioral Therapy.* Department of Veterans Affairs South Central MIRECC, Houston, 2008.

[2] J. Calvo. 2022. *Aprendizaje por transferencia: NLP Blog Europeanvalley. [online] Europeanvalley. es. Available at:* *https://www.europeanvalley.es/noticias/aprendizaje-por-transferencia-nlp/* *[Accessed*, 26, September 2022.

[3] Introduction — Machine Learning — Google Developers — developers.google.com. https://developers.google.com/machine-learning/guides/text-classification. [Accessed 26-Sep-2022].

[4] MA. Courtney E. Ackerman. CBT Techniques: 25 Cognitive Behavioral Therapy Worksheets — positivepsychology.com. https://positivepsychology.com/cbt-cognitive-behavioral-therapy-techniques-worksheets/. [Accessed 27-Sep-2022].

[5] Asphalion - Scientific and Regulatory Affairs consultancy — asphalion.com. https://www.asphalion.com. [Accessed 06-Dec-2022].

[6] Rohith Gandhi. Support Vector Machine — Introduction to Machine Learning Algorithms — towardsdatascience.com. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47. [Accessed 01-Jan-2023].

[7] Payscale - Salary Comparison, Salary Survey, Search Wages — payscale.com. https://www.payscale.com. [Accessed 06-Oct-2022].

[8] Mental health — who.int. https://www.who.int/health-topics/mental-health#tab=tab_1. [Accessed 25-Sep-2022].

[9] The treatment gap in mental health care - PubMed — pubmed.ncbi.nlm.nih.gov. https://pubmed.ncbi.nlm.nih.gov/15640922/. [Accessed 11-Jan-2023].

[10] Michiyo Hirai and George A. Clum. A meta-analytic study of self-help interventions for anxiety problems. *Behavior Therapy*, 37(2):99–111, June 2006.

[11] Benjamin Shickel, Scott Siegel, Martin Heesacker, Sherry Benton, and Parisa Rashidi. Automatic detection and classification of cognitive distortions in mental health text. *Automatic Detection and Classification of Cognitive Distortions in Mental Health Text*, 9 2019.

[12] Ignacio de Toledo Rodriguez, Giancarlo Salton, and Robert Ross. Formulating automated responses to cognitive distortions for cbt interactions. 2021.

[13] Daniel M Low, Laurie Rumker, John Torous, Guillermo Cecchi, Satrajit S Ghosh, and Tanya Talkar. Natural language processing reveals vulnerable mental health support groups and heightened health anxiety on reddit during covid-19: Observational study. *Journal of medical Internet research*, 22(10):e22635, 2020.

[14] Degree final project — FIB - Barcelona School of Informatics — fib.upc.edu. https://www.fib.upc.edu/en/studies/bachelors-degrees/bachelor-degree-informatics-engineering/degree-final-project. [Accessed 29-Nov-2022].

[15] Writing a GDPR-compliant privacy notice (template included) - GDPR.eu — gdpr.eu. https://gdpr.eu/privacy-notice/. [Accessed 05-Dec-2022].

[16] Art. 13 GDPR - Information to be provided where personal data are collected from the data subject - GDPR.eu — gdpr.eu. https://gdpr.eu/article-13-personal-data-collected/. [Accessed 05-Dec-2022].

[17] Chapter 3 (Art. 12-23) Archives - GDPR.eu — gdpr.eu. https://gdpr.eu/tag/chapter-3/. [Accessed 05-Dec-2022].

[18] CO2 emissions per kWh in Spain - Nowtricity — nowtricity.com. https://www.nowtricity.com/country/spain/. [Accessed 10-Jan-2023].

[19] API Reference — scikit-learn.org. https://scikit-learn.org/stable/modules/classes.html#module-sklearn.naive_bayes. [Accessed 01-Jan-2023].

[20] Why 85 https://www.techrepublic.com/article/why-85-of-ai-projects-fail/. [Accessed 13-Jan-2023].