FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

HandsDown: Data Entry Automation During Office Visits

Diogo Afonso Duarte Reis

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Liliana Ferreira

August 6, 2020

HandsDown: Data Entry Automation During Office Visits

Diogo Afonso Duarte Reis

Mestrado Integrado em Engenharia Informática e Computação

August 6, 2020

Resumo

Apesar dos computadores facilitarem o processo, o preenchimento dos registos de saúde eletrónicos durante e após as consultas com doentes, ainda é a tarefa mais vezes identificada como morosa e complexa. É incómoda, demorada e está sujeita a erros. Os médicos utilizam os computadores para não terem de escrever registos médicos apenas em papel, poupando tempo e garantindo uma maior legibilidade para toda a equipa médica que os for ler. No entanto, esta transição do papel para o computador trouxe novos desafios. Para que os médicos trabalhem de uma forma mais rápída e eficaz, eles passam uma boa parte das suas consultas a olhar para o computador, escrevendo os pedaços relevantes da conversa com os seus doentes. Isso faz com que o doente pense que não está a ser ouvido sentindo indiferença, alienação e desinteresse por parte do médico.

Essa insatisfação serve como motivação ao desenvolvimento de um sistema automatizado capaz de transcrever diálogo entre médico e doente e criar uma nota clínica, estruturando as transcrições. O objetivo é ser possível realizar reconhecimento de voz, em português europeu, durante as consultas, utilizando o microfone de um dispositivo móvel e um serviço online de transcrição de voz para texto. Técnicas de processamento de linguagem natural serão depois utilizadas para criar uma nota clínica a partir das transcrições. Como consequência, o médico poderá interagir mais eficazmente com o doente e utilizar melhor o seu tempo.

A nota médica SOAP, abreviatura para Subjetivo, Objetivo, Avaliação e Plano, e estrutura usada por profissionais médicos durante documentação clínica, permitiu o desenvolvimento de um modelo de aprendizagem automática capaz de recolher frases de transcrições de diálogo e categorizar cada uma com uma das quatro categorias SOAP.

Para treinar e testar a nossa solução, um conjunto de textos médicos, disponíveis de forma pública, foi agregado e cada frase foi manualmente categorizada com uma das quatro categorias SOAP. Os resultados foram positivos e permitiram antecipar que a solução conseguirá ainda melhores resultados na presença de um conjunto de dados mais vasto.

Palavras-Chave: Automação de dados, Processamento de linguagem natural, Reconhecimento automático de fala

Abstract

Alongside the ever-increasing applications of technology, computers were eventually integrated into medical rooms to facilitate the process of documentation work and ease doctors' burdens. Instead of having doctors write medical records by hand and dealing with them physically after each patient's appointment, they use a computer to write medical notes of each appointment, saving a lot of time in the process. However, this change from paper to digital promoted a new problem. For the physician to work faster and more efficiently, he will spend most of the consultation looking at the computer, writing down the required bits of conversation between him and the patient. As a consequence, the patient feels indifference, alienation and frequent reports feelings of not being listened by the medical professionals.

Despite having computers ease the process, filling out the electronic health record during and after patient visits is still one task physicians often complain about, being disruptive, error-prone, tedious, and time-consuming. This complaint will serve as the main problem to be addressed with the development of a fully automated pipeline capable of transcribing dialogue and creating medical notes from the resulting transcriptions. Its purpose will be to perform automatic speech recognition, in European Portuguese, during appointments, using a mobile phone's built-in microphone and online service speech-to-text service. Natural Language Processing techniques will then act on the resulting text and create a summarized medical note. As a consequence, the physician would interact more effectively with the patient while better utilizing his time.

The SOAP format, which stands for Subjective, Objective, Assessment and Plan, was found to be a standard structure, used by health professionals for clinical documentation, that allowed for the development of a fine-tuned machine learning model capable of analyzing sentences from speech transcriptions and categorizing each one with one of the four SOAP categories.

To train and test our solution, a dataset of publicly available medical data was collected and each sentence manually labeled with one of the four SOAP categories. The results were positive and allowed to anticipate that the solution can provide even more accurate results with a larger dataset.

Keywords: Clustering and classification, Natural Language Processing, Automatic Speech Recognition

Agradecimentos

Agradeço à minha orientadora Liliana Ferreira pelo apoio ao longo da realização desta dissertação.

Obrigado aos meus amigos pelas discussões interessantes que tivemos àcerca da dissertação e de todo o apoio e ajuda que ofereceram.

E agradeço profundamente à minha família, pelas ideias que me transmitiram, pela motivação que me deram e por todo o trabalho que tiveram de realizar para permitir que tivesse o melhor ambiente de trabalho durante o confinamento.

Diogo Reis

"Ki-woo: Dad?
Ki-taek: Yeah?
Ki-woo: What was your plan?
Ki-taek: What are you talking about?
Ki-woo: Before you said you had a plan. What will you do about the basement?
*Ki-taek: Ki-woo, you know what kind of plan never fails? No plan at all. No plan. You know why? If you make a plan, life never works out that way. Look around us, did these people think 'Let's all spend the night in a gym?' But look now, everyone's sleeping on the floor, us included. That's why people shouldn't make plans. With no plan, nothing can go wrong and if something spins out of control, it doesn't matter. Whether you kill someone or betray your country. None of it f*cking matters. Got it?*
Ki-woo: Dad, I'm sorry.
Ki-taek: For what?
Ki-woo: Everything."

'Parasite' (2019)

Contents

List of Figures

List of Tables

Abbreviations and Symbols

API	Application Programming Interface
App	Application
ASR	Automatic Speech Recognition
BERT	Bidirectional Encoder Representations from Transformers
CD	Clinical Documentation
CEO	Chief executive officer
CSS	Cascading Style Sheet
eHealth	Electroning Health Services
EHR	Electronic Health Record
Grus	Gated Recurrent Unit
HTML	HyperText Markup Language
LSTMs	Long short-term memory networks
MT	Machine Translation
NER	Named Entity Recognition
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
QA	Question Answering
RBMT	Rule Based Machine Translation
SMT	Statistical Machine Translation
SOAP	Subjective Objective Assessment Plan
UMLS	The Unified Medical Language System
UMLFiT	Universal Language Model Fine Tuning
USA	United States of America
WER	Word Error Rate
WWW	World Wide Web

Chapter 1

Introduction

This document is a part of the development of the Master's thesis in Informatics and Computing Engineering, being carried out in partnership with Fraunhofer Portugal Research Association. This chapter provides an overview of the context and motivation for the dissertation. The research objectives and a brief overview of the methodology are also presented.

1.1 Context

Electronic health records (EHR) play a crucial role in the short and long-term coordination of patient care, public health, and clinical medical research (Krishna et al., 2020). They can solve a plethora of challenges currently faced in the healthcare industry, leading to the potential to decrease costs, improve quality of care, and the ability to manage resources (Andrade, 2017).

EHRs can be defined as a group of varied types of documents, including discharge summaries, medical history, allergies, pathology reports, laboratory results, (Rajput et al., 2017), patient demographics, physician notes, nursing assessments, and nursing orders (Adler-Milstein et al., 2017), that relate to an individual. If a comprehensive EHR is implemented, it can also provide support for clinical reminders, clinical guidelines, drug-drug interactions, drug allergy results, drug dosing support, and drug-lab interactions (Adler-Milstein et al., 2017).

The EHR allows for the generation of complete records of patient encounters, in addition to supporting other care-related activities. These include evidence-based decision support, outcomes reporting, and quality management (Fareed et al., 2015).

1.1.1 EHR Adoption in Healthcare

There have been initiatives driven by the government of the United States of America (USA), for the implementation of EHR services (Abramson et al., 2012). This led to an increase of 65% in the adoption of EHRs by office-based physicians between 2004 and 2017 (Office of the National Coordinator for Health Information Technology, 2019), as seen in the table 1.1, and a total of approximately 87% of hospitals had adopted some form of EHR in 2015 (Adler-Milstein et al., 2017).

	2004	**2007**	**2011**	**2014**	**2017**
Office-based physicians	20.8%	34.8%	57%	82.8%	85.9%

Table 1.1: Adoption of EHR technology in the USA

In Portugal the adoption of electronic health systems began in the 1990's where information systems were designed that would support the management and control of the flow of users, the standardization of clinical and administrative data, and the enabling of automatic billing and improvement in the communication between health providers (Catan et al., 2015). In Portugal, according to the National Statistics Institute, 93% of the official's hospitals and 73% of the private hospitals have an Electronic Medical Record (EMR) (Catan et al., 2015), which still is a digital record of a patient such as an EHR, but is specific to each hospital and does not get shared around between different medical offices.

1.2 Motivation

Healthcare organizations are constantly being pressured into implementing EHR technologies, because of administrative, clinical, financial, and regulatory demands, even in the presence of weak technical infrastructures (Fareed et al., 2015). One of the ways these organizations try to provide high-quality services while reducing costs and improving performance is through medical documentation in EHRs. However, populating the data in EHRs places a massive burden on healthcare providers (Krishna et al., 2020), forcing them to dedicate an unreasonably high amount of time to fill in these documents. It was found that during Cardiology appointments, physicians only spend 27% of the consultation's time interacting with their patients, while 49% of the time is spent with electronic documentation (Jadczyk et al., 2019). Studies also show that for every hour that physicians spend seeing patients, they spend about 45 minutes on EHR documentation (Krishna et al., 2020) which often is not enough to complete their documentation work resulting in overtime hours, which can sometimes reach 2 hours (Sinsky et al., 2016). Additionally, when physicians document visits too long after their conclusion, imperfect recollection may result in noncomprehensive or even erroneous documentation. Hence, automatic systems that improve the efficiency of EHR documentation can potentially mitigate a critical pain point in the medical profession (Krishna et al., 2020).

All of the previous factors contribute to worsening the burnout most physicians feel due to an overload of work. They feel emotionally fatigued and depersonalization, including cynicism and a lack of empathy, leading to decreased motivation, increased dissatisfaction, and a low sense of personal accomplishment (Beresford, 2016).

The economic impact of the physician burnout is also enormous - a recent study from the Annals of Internal Medicine estimated that physician turnover and reduced clinical hours due to burnout account for approximately $4.6B in costs each year (Han et al., 2019).

Figure 1.1: Least satisfying factors about medical practice (Suki, 2019)

Poorly-designed EHR systems also create increasing amounts of administrative work for physicians to document patient encounters, quality measures, and other requirements (Brita Belli, 2019), and are also one of the aspects most physicians complain about as seen in figure 1.1. As such, speech systems in conjunction with advances in Natural Language Processing and Machine Learning, that use speech recognition to understand human speech, and use the recorded information to accomplish the users' needs, have led to breakthroughs in the healthcare industry (Velupillai et al., 2018; Lee et al., 2019). These systems lead to more efficient services' delivery and help improving the quality of documentation, while increasing productivity, without impacting the physicians in any negative way (Ajami, 2016).

1.3 Research Objectives

The main goal to be accomplished is the development of an automated computational pipeline for the transcription of medical appointments in European Portuguese and the creation of medical notes, that takes advantage of state-of-the-art Speech-to-Text solutions as well as advances in NLP and Deep Learning. The focus of this dissertation is to study if an automatic note generation pipeline is achievable. Due to the lack of comparable data to analyze the developed solution, the higher the accuracy the more acceptable the solution becomes.

Furthermore, the previously mentioned objectives are meant to be accomplished without degrading the patient-physician relationship while also providing an optional and non-intrusive system.

Gathering medical data for study and processing is also a challenge that was targeted. Since most text processing systems rely heavily on the quality and quantity of data, another research objective was defined. The goal is to collect publicly available datasets based on real medical notes and dialogue and study their quality and determine if the quantity gathered is enough for the development of the pipeline.

The accuracy of Automatic Speech Recognition (ASR) online services for the transcription of medical information in European Portuguese will be studied alongside the pipeline's development.

The resulting text from the pipeline will also be evaluated based on how accurate it can label spoken sentences into each of the following categories, "Subjective", "Objective", "Assessment"

and "Plan" that together produce a medical note in the SOAP format, which will be thoroughly detailed in the following section and chapters.

The research question that will be explored is what impact can the use of Natural Language technology interfaces, that support the filling of EHRs, have on the patient-physician relationship?

1.4 Methodology and Expected Results

As mentioned previously the final goal is for a newly created pipeline to be able to create medical notes from the transcription of spoken dialogue between a physician and a patient. These transcriptions are raw unstructured text, thus in order to accomplish the note generation, it is useful to follow a specific type of structure. This is where the SOAP medical format fits in. After seeing a patient, doctors typically document the encounter in SOAP notes, semi-structured written accounts containing four sections: (S)ubjective information reported by the patient; (O)bjective observations, e.g., lab results; (A)ssessments made by the doctor (typically, the diagnosis); and a (P)lan for future care, including diagnostic tests, medications, treatments, and follow-up protocol. Each section is further divided into subsections giving it a finer substructure. For example, the subjective section contains 9 subsections, e.g., chief complaint and past medical history. A visit may not have information relevant to each subsection, and thus some of the subsections may be empty. The fraction of times a subsection is populated varies widely: the team at (Krishna et al., 2020) found that allergies are the sparsest (present in about 4% of notes), the chief complaint is the most frequently observed (present in every note). Figure 1.2 represents a part of a sample of a medical note in the SOAP format.

SUBJECTIVE: The patient is a 49-year-old white female, established patient to Dermatology, last seen in the office on 08/10/2004. She comes in today for reevaluation of her acne plus she has had what she calls a rash for the past two months now on her chest, stomach, neck, and back. On examination, this is a flaring of her acne with small folliculitis lesions. The patient has been taking amoxicillin 500 mg b.i.d. and using Tazorac cream 0.1, and her face is doing well, but she has been out of her medicine now for three days also. She has also been getting photofacials at Healing Waters and was wondering about what we could offer as far as cosmetic procedures and skin care products, etc. The patient is married. She is a secretary.

FAMILY, SOCIAL, AND ALLERGY HISTORY: She has hay fever, eczema, sinus, and hives. She has no melanoma or skin cancers or psoriasis. Her mother had oral cancer. The patient is a nonsmoker. No blood tests. Had some sunburn in the past. She is on benzoyl peroxide and Daypro.

CURRENT MEDICATIONS: Lexapro, Effexor, Ditropan, aspirin, vitamins.

PHYSICAL EXAMINATION: The patient is well developed, appears stated age. Overall health is good. She has a couple of acne lesions, one on her face and neck but there are a lot of small folliculitis-like lesions on her abdomen, chest, and back.

IMPRESSION: Acne with folliculitis.

TREATMENT:
1. Discussed condition and treatment with the patient.
2. Continue the amoxicillin 500 mg two at bedtime.
3. Add Septra DS every morning with extra water.
4. Continue the Tazorac cream 0.1; it is okay to use on back and chest also.
5. Referred to ABC clinic for an aesthetic consult. Return in two months for followup evaluation of her acne.

Figure 1.2: Example of a medical note in the SOAP format (MTHelpLine)

For these goals, a mobile application for Android devices was developed. Its purpose is to capture audio during medical appointments using the internal microphone and, in near real-time, relying on external Google's Cloud Speech-to-Text API (IBM, 2020b), transcribe that audio into text.

During and after the development of the pipeline, it is expected to evaluate its efficiency, which will be based on the accuracy of the creation of medical notes that will be based on the categorization of sentences in the collected transcriptions.

1.5 Document Structure

The rest of the document follows this structure: chapter 2 provides an in-depth review of the systems used by medical personnel such as the Electronic Health Record, it details what a medical note is and why the SOAP format is used and important. This chapter also provides information about clinical documentation, state-of-the-art speech recognition solutions, a detailed NLP pipeline prototype, and some relevant applications of NLP. It ends with a section dedicated to SOAP classification which the main goal to accomplish for the note generating. Chapter 3 discusses the two layers that are involved in the making of the pipeline for the automatic generation of medical notes and which technologies they use. Chapter 4 details the architecture of the implemented solution, by going into detail into what the mobile app can accomplish and that the server and website can accomplish. In chapter 5 is where the solution is evaluated and chapter 6 presents a summary of everything that was accomplished, some conclusions, limitations, and future work.

Chapter 2

Literature Review

2.1 Health Record

A patient's health record is a set of different types of clinical information, gathered from different sources, associated with a patient's mental and physical health. Health records contain demographic data, next of kin, and some of the following: diagnoses, treatments, biological exams, examinations, imaging exams, allergies, nursing records, referrals for treatment, consent forms for surgical procedures, discharge letters, post-mortem reports among others (Segen's Medical Dictionary, 2011).

2.1.1 Electronic Health Record (EHR)

Taking the patients' health records and making them available outside of the restrictions of physical paper is what an Electronic Health Record is responsible for. EHRs are equivalent to digital medical records of patients, that are easily and securely accessible by authorized personnel. Besides containing the previously mentioned data such as laboratory and test results and medical history, EHRs also allow access to evidence-based tools that providers can use to make decisions about a patient's care and automate and streamline provided workflow (HealthIT, 2019).

A major benefit when using EHRs is that, with proper authorization, a medical provider can create and manage health information and share it with other providers across multiple health care organizations (HealthIT, 2019). EHR may also be frequently mistaken for EMR and vice-versa, as both of them are digital records of patient health information. This is because there is only a slight difference between both terms(NextGen, 2020). EMR digital data usually stays in the doctor's office and does not get shared, while EHRs were built to be able to share information with multiple healthcare providers, as they contain all the information written by the different medical personnel involved in a patient's care (NextGen, 2020; Faizan, 2020; HealthIT, 2019).

2.1.2 Clinical Documentation

Clinical Documentation (CD) is the process of generating medical notes after every patient encounter that will be inserted into an EHR system or written in a paper. The documentation captures patient care from admission until discharge and should always be done with the highest level of responsibility, no matter the severity of the case. Clinical documentation is a fundamental skill, that when mastered can provide accurate, thorough clinical documentation that benefits patients, healthcare providers, and healthcare facilities (Myrick, 2019; Faizan, 2020; Sando et al., 2017).

Good documentation is not just important for legal reasons it can also impact the quality of patient care and even hospital funding. However, writing good documentation can become a low-priority task. The following reasons adapted from (Syed, 2020) explain why good documentation should be prioritized:

1. **It is a form of communication**
   Writing good documentation, during the care of a patient, is essential, since it properly informs all the involved medical personnel about the patient's current situation. This ensures the best possible quality of care for the patient and avoids negative impacts caused by poor documentation.

2. **It is a legal document**
   Good documentation can also affect the outcome of any legal proceedings. When this happens, since all of the medical records are permanent and accessible, they will be heavily scrutinized to find any gaps in someone's argument or help an individual to support their argument.

3. **It is a document of service**
   This is a point often missed. Medical documentation is a document of service that has huge implications for hospital funding. Each issue that is documented is coded and then translated into a cost for the hospital system. Thorough documentation of all medical issues and treatments is therefore crucial for hospital funding, particularly in discharge summaries.

Medical institutions can adopt three different modes of documentation: writing notes by hand, typing on computers, or using speech recognition to capture the physicians' voices. One mode does not negate another mode but, as it will be discussed below, each one presents their challenges:

- **Handwritten**
  In clinical records, many notes are still being written by hand which can make them harder to read. The author may understand what he wrote, but when a third party gets hold of the author's notes and tries to read them, problems can arise. One study that examined clinical histories samples from a Spanish General Hospital found that 18 out of 117 case notes, 15%, "were so illegible that the meaning was unclear" (Rodriguez-Vera et al., 2002). Apart from the legibility of the physicians' handwriting, paper is also a physical item that can be easily lost or damaged from the environment it is kept on.

- **Typed on a computer**
  With the integration of EHRs in the medical rooms comes the ability for medical personnel to type into a computer their medical notes instead of having to write them by hand. This reduces the usage of paper and ensures a method to which every third party medical staff can easily read what was written. However, physicians still end up doing up to 2 hours of overtime documentation work, even with the usage of a computer (Sinsky et al., 2016).

- **Voice oriented documentation**
  Being able to use a person's voice to create medical notes is another method for documentation. Using ASR technologies a physician can talk into a microphone, see the transcription performed on his speech, and edit as he sees fit. This approach also has its drawbacks since manual typing will always be more accurate than a speech transcription and the right equipment is also needed.

There are currently some challenges that need to be tackled by healthcare providers and it is a major difficulty to try and overcome them all. One problem is the low quality of clinical reports and high time consumption. When there is a lack of time from the physician, due to all of the motives presented in the motivation section 1.2, it is expected that the medical notes will not be as accurate as they could be with proper time and care. Another problem is the loss of minor clinical encounters, which, due to the overload of documentation work that physicians can experience, minor encounters are often ignored and provide a risk to the quality of healthcare delivery. High costs is another problem since by implementing complex EHR systems, or by employing dedicated transcriptionists to help the physicians' workload, the healthcare providers will often require extensive funds (Faizan, 2020).

2.1.3 SOAP Medical Note

A medical note is an entry into a medical or health record made by a member of a patient's healthcare team during a patient's visit or outpatient care (Seo et al., 2016). These notes are used to facilitate the understanding of the current state of patients as well as improving communication between the professionals involved in the care of the patient so that everyone can carry on the work from where the last person left off (Practicefusion, 2020; Brock, 2016). A medical note will be one of a list of different types, such as discharge summaries, patient demographics, physician notes, nursing assessments, and nursing orders (Adler-Milstein et al., 2017) and will follow a standard structure and guidelines such as the SOAP format or will be unstructured (Faizan, 2020).

The SOAP format is a widely used method of clinical documentation, theorized by Larry Weed almost 50 years ago, that allows for healthcare providers to share information in a universal, systematic and easy to read format (Podder V, Lew V, Ghassemzadeh S., 2020; Continuum, 2020). Each category is described below, adapted from (Podder V, Lew V, Ghassemzadeh S., 2020):

- **Subjective**
  When documenting in this section the physician will take into account the "subjective" experiences of a patient. Therefore, the patient's personal feelings or views, or that of someone close to him will be written here by the physician. This section will provide context for the "Assessment" and "Plan" categories.

- **Objective**
  This section documents every factual and objective data collected by the physician during his interaction with a patient. These include vital signs, such as temperature and heartbeat, observations of a person's behavior, laboratory data such as white blood cell count, imaging results, and review of past medical interactions.

- **Assessment**
  The assessment section is where the physician's thoughts on the salient issues and the diagnosis (or differential diagnosis) are written, which will be based on the information collected in the previous two sections (Potter·Documentation, 2019).

- **Plan**
  Every step that is being taken for the care of a patient is written in this section. These include the need for consulting with other clinicians and for additional testing. This section can also provide the patient's next clinician a way for him to understand what needs to be done.

Writing a SOAP note has some benefits such as helping healthcare workers use their clinical reasoning to assess, diagnose, and treat a patient based on the information provided by the SOAP notes (Podder V, Lew V, Ghassemzadeh S., 2020) By following a defined structure, SOAP notes can lower the efforts required to extract desired information efficiently and quickly (Belden et al., 2017; Faizan, 2020). SOAP notes are clear, accurate, and concise and allow for the efficient sharing of information between healthcare providers. Due to this, providers explicitly use there reports to give recommendations to each other (Seo et al., 2016; Faizan, 2020; Lisenby et al., 2018). By having specific tasks while writing a medical note, medical personnel are encouraged to complete their reports and ensure that the quality of their work also enhances the quality of healthcare delivery (Lenert, 2017; Faizan, 2020). SOAP reports also keep the records of the encounters between patients and healthcare professionals, meaning that they can serve as an evaluation tool for accountability, billing, and legal documentation (Colicchio and Cimino, 2018; Faizan, 2020; Mathioudakis et al., 2016; Pearce et al., 2016).

Despite the benefits of following the SOAP structure, there have also been discussions about the best order to the categories. For instance, ordering the structure by having the last two categories first and the first two categories in the end(APSO), it was determined that there was an increase in speed and task success (Podder V, Lew V, Ghassemzadeh S., 2020). Another SOAP's weakness is the lack of time accountability, meaning there is no way to document changes over time. The SOAPE format is an alternative that tries to fill in the gap left by not taking time into account, with the letter E being a reminder to medical personnel to write changes over time and

assess how well the plan defined in the previous section worked (Podder V, Lew V, Ghassemzadeh S., 2020). The SOAP format has also been criticized for its encouragement of using difficult to decipher medical information as well as overusing abbreviations and of making health professionals simply collect information and not assess it (Physiopedia, 2019).

2.2 Speech Recognition

Speech is one of the most important needs in modern civilized societies and the most convenient means of communication between people. Speech recognition refers to the process of collecting speech and processing it into text. When there is an exchange of information between people who do not speak each other's language, language technologies are there to provide solutions for this problem in the form of ordinary interfaces so that the information can be understood by everyone (Reddy and Mahender).

Due to major companies such as Google, Microsoft, IBM, Apple, and Amazon, exploring the field of Automatic Speech Recognition has become a very enticing prospect. The development of intelligent personal assistants, such as Amazon's Amazon Alexa, Apple's Siri or Google's Google Assistant, which rely on Users' Speech and its transcriptions to text to perform their desired actions and help them with day-to-day activities, has managed to make this, a relevant area to explore over the years. The field has benefited from advances in deep learning and big data, boosting the performance of the algorithms used on ASR systems to very reliable standards (Hinton et al., 2012).

The development of an ASR system is a complex endeavor that requires thousands of annotated training data and immense computational power to provide a usable translator (Hinton et al., 2012). With this in mind, this dissertation will focus on using existing APIs, which allow the use of highly trained and complex models that transcribe voice data into text.

2.3 Speech Recognition Systems in the Health Care Industry

Currently, in the healthcare industry, there is a desire to explore how speech-to-text systems can improve the lives of physicians and patients, and increase the value of the organizations that support them(Ajami, 2016). One of the ways offices are trying to improve their delivery of services is in the documentation sector. As mentioned before in chapter 1, the documentation performed by physicians leaves them exhausted and unmotivated because of their workload, which in turn leads to poor delivery of services(Beresford, 2016). The following systems were developed to help reduce these issues:

- **CardioCube** (Jadczyk et al., 2019)
  CardioCube is a voice-enabled technology that combines deep learning and natural language understanding to implement human-machine verbal communication into the clinical field to optimize organizational performance and workflow.

The CardioCube team did a study with 10 males and 12 females, with 9 of them diagnosed with cardiovascular disease. While utilizing a patient registration software implemented on an Amazon Echo and a web-based EHR system, the participants verbally answered some clinical questions.

The results were optimal in relation to the verbally provided data and the system was able to generate a summarized medical report and instantly accessible in the web-based EHR system.

- **Amazon Transcribe Medical** (Amazon, 2020a)
  Amazon Transcribe Medical is a service based on Amazon Comprehend Medical that uses machine learning algorithms to accurately capture a dialogue between a physician and a patient and transcribe it into text. Transcribe Medical uses machine learning to provide highly accurate automatic speech recognition (ASR) for the medical industry.

- **Nuance's ambient clinical intelligence** (Nuance, 2020)
  The ambient clinical intelligence system from Nuance offers a voice-enabled solution that can be used during an appointment between a physician and a patient to automatically document their dialogue into text. The purpose of this system is for the physician to reduce the time he spends on the computer during an appointment so that he can focus more on the patient and make him feel heard.

- **Suki AI** (Suki, 2020)
  Suki is an AI-driven voice-enabled digital assistant for doctors, available on iOS devices, to alleviate administrative work like medical charting in EHRs. The mobile application works by having doctors speak into the phone's microphone. With the collected speech the app can either perform an action such as creating a new note or transcribe the whole speech into a medical note.

  As claimed by Suki, the mobile application can lower the amount of time spent on documentation work by 76% while also providing high-quality medical notes that decrease the amount of denied notes by 19% (Pennic, 2020).

- **HealthTalks** (Monteiro and Lopes, 2018)
  The HealthTalks mobile application, despite being directed to physicians, still proves to be relevant to this dissertation, for its use of automatic speech recognition in the medical domain. The app works by having the user open it during an appointment with his physician, the dialogue between the two is recorded and later transcribed into text and the most important medical concepts are then displayed along with their definitions. The goal of the app is for the user to have more control over their health condition by providing patients with a tool to help them overcome their low health literacy levels.

  The speech recognition service that was used for this app was Google's Cloud Speech API, reaching an error rate of 12 percent in medical texts which was superior to the alternative Bing Speech API.

2.4 Automatic Speech Recognition Online Services

There are multiple Automatic Speech Recognition or ASR services but since Portuguese will be the focus of this dissertation, the services that do not support it were automatically discarded. By filtering out the services that do not provide Portuguese support, four possible candidates remain:

- Google's Cloud Speech-to-Text (IBM, 2020b)
- Microsoft Azure Speech-to-Text (Microsoft, 2020)
- Amazon transcribe (Amazon, 2020b)
- IBM Watson Speech-to-Text (IBM, 2020a)

Despite supporting the Portuguese language, IBM Watson only supports Portuguese from Brazil(IBM, 2020a) whereas the other three support European Portuguese (IBM, 2020b; Microsoft, 2020; Amazon, 2020b). This leaves IBM Watson as the last possible candidate. Determining which of the remaining services is the best will be discussed in the following sections.

2.4.1 Performance of Automatic Speech Recognition Services

Determining which service most accurately transcribes speech into text is the first step into finding which service is best. A frequently used metric to compare performance between services is the Word Error Rate or WER(Costa, 2019). WER is defined on Equation 2.1, where S is the number of substitutions performed, D the number of deletions, I the number of insertions, N the number of spoken words in the original reference, and C is the number of correctly guessed words (Costa, 2019).

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \tag{2.1}$$

A study conducted in 2019 with 15 males and 15 females concluded that the average WER for Google's Cloud Speech-to-Text and Microsoft Azure Speech-to-Text was 12.29% and 17.67% respectively (de Toledo et al., 2019). A study conducted in 2018 concluded that Google's Cloud Speech-to-Text had a WER 20 to 24% lower than Microsoft Azure Speech-to-Text (Monteiro and Lopes, 2018)

With these results, we can determine that Google's Cloud Speech-to-Text API is the most accurate solution and the best possible candidate for implementation in the solution.

2.5 Natural Language Processing

Natural Language Processing, normally abbreviated as NLP, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language. A definitive goal of NLP is to read, decipher, comprehend, and understand the human dialects in a way that is significant (Garbade, 2018a; Pinto, 2015). NLP techniques began by being rule-based,

with individualized conditions for every specific case present in a given language, creating a set of heuristics for processing text. More recently, NLP bases itself on language models that rely on statistical inference over large text corpora. With the advent of Deep Learning NLP models, the tools of this area are being ever more robust and powerful to deal with the immense variability of written text, languages, and contexts (Steven Bird and Loper, 2009).

Each step of a robust NLP system is presented in figure 2.1 and explored below:

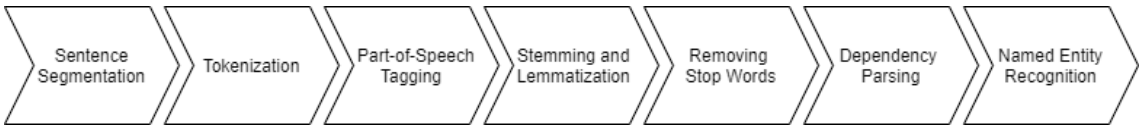

Figure 2.1: NLP pipeline (Geitgey, 2019)

2.5.1 Sentence Segmentation and Word Tokenization

Understanding each sentence instead of the whole text at once is a lot easier (Geitgey, 2019). Separating sentences when a punctuation mark is found can be a simple way of solving this step. However, it doesn't always work, for example when "Mr. John" is used, the system shouldn't separate "Mr" into its separate sentence. To avoid this problem, tables of abbreviations that contain periods can help prevent these types of incorrect sentence boundaries (Kapetanios et al., 2013).

After the sentences are separated, the system can analyze each one and break them into words or tokens, using the space between words to split them into different words (Geitgey, 2019). The results are shown in figure 2.2.

2.5.2 Part-of-Speech Tagging

Part-of-Speech tagging consists of labeling each of the resulting words from the previous step with their appropriate part of speech, which includes verbs, nouns, adjectives, adverbs, conjunction, pronouns, and their sub-categories (PAR).

Each token needs to be assigned to their correct part of speech and to do this, each word is analyzed by a pre-trained part-of-speech classification model. By processing some words from the first sentence, this is the outcome:

- "London" - Proper Noun
- "is" - Verb
- "the" - Determiner
- "and" - Conjunction
- "populous" - Adjective
- "of" - Preposition

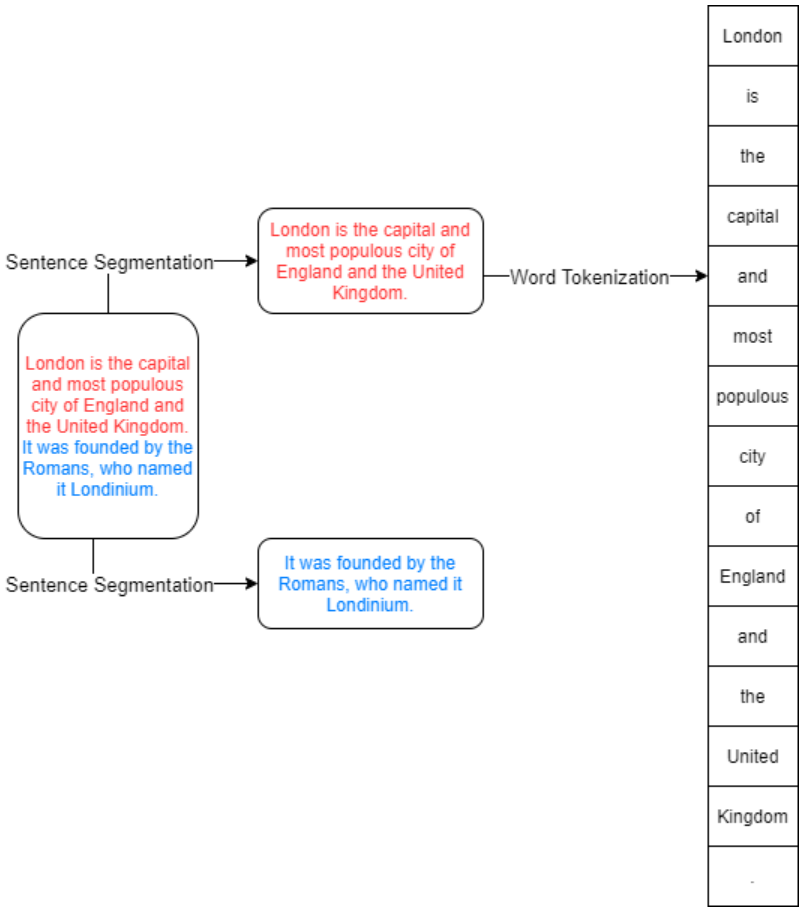

Figure 2.2: Sentence Segmentation and Work Tokenization

2.5.3 Stemming and Lemmatization

For grammatical reasons, pieces of text will use different inflections for the same word, for example, "organize", "organizes", and "organizing". One word can have multiple variations according to size, gender, and number (STE, 2008).

The goal of both of them is to reduce inflectional forms and achieve a normalized base related to the different inflections. For example:

- "am, are, is" -> be
- "car, cars, car's, cars'" -> car

However, Stemming and Lemmatization differ from one another by the process for which they achieve the previous results (STE, 2008).

The process of stemming differs from lemmatization in the way it uses a crude heuristic process to remove the suffixes of words to achieve the previous results (STE, 2008). The most commonly used algorithm to perform stemming in Portuguese text is "Removedor de Sufixos da Língua Portuguesa" (RSLP) that produces the following results(Viera, A.F.G. and Virgil, J., 2007):

- "balões" -> "bal"

- "aviões" -> "avi"
- "avião" -> "avi

As we can see this process works with the words "aviões" and "avião" which result in the same word "avi".

Lemmatization is a more complex method than stemming, in the way it uses a tool called "lemmatizer" that does full morphological analysis to accurately identify the lemma for each word (STE, 2008). The lemma of a word is the form of that word that would be found in a dictionary. This way lemmatization differs from stemming in the way that the result is always an intelligible word with meaning.

By using lemmatization in the resulting words after Part-of-Speech tagging, the word "is" after "London" is changed to "be".

2.5.4 Removing Stop Words

In the resulting text from the previous steps, there can be found filler words that appear frequently which do not add anything of value to the text itself. Words such as "and", "the" and "a" for example, if they are not used for naming a band or movie. These words are considered **stop words**, that can be filtered out before doing any statistical analysis, by checking a hardcoded list of known stop words (Geitgey, 2019).

Analyzing the resulting sentence from the previous steps we can remove "is", "the", "and" and "most".

2.5.5 Named Entity Recognition (NER)

The goal of NER is to detect and label nouns with the real-world concepts that they represent. NER systems should tag people's names, company names, geographic locations, product names, dates and times, amounts of money, names of events (Geitgey, 2019). In the first phrase, "London" is considered a Geographic Location.

By analyzing the phrase "Google's CEO Sundar Pichai introduced the new Pixel3 in New York" it can retrieve "Google" as an Organization, "Sundar Pichai" as a Person, and "New York" as a Location.

2.5.6 Word Ontologies

Ontologies were developed in Artificial Intelligence to facilitate knowledge sharing and reuse (Fensel, 2001). They are a representation of words of a given field, with their associated definition and possible relations with other words or concepts of said field (Costa, 2019).

Biomedical data is critical to building accurate NLP systems. This is why, in 1986, the Unified Medical Language System (UMLS) started being developed. The purpose was for it to contain a collection of medical concepts and their relations. Currently in the Metathesaurus there over 1 million concepts and 5 million concept names originating from over 150 controlled vocabularies

in the biomedical sciences (Doan et al., 2014). The different concepts are also categorized by their semantic types, leading to 135 categories and 54 relationships among categories (Doan et al., 2014). The UMLS also provides domain models and trained corpora that target specific domains such as radiology reports and contain manually labeled text used for machine learning (Doan et al., 2014).

2.5.7 Word Embeddings and Deep Learning

In almost every recent NLP model there is a chance that word embeddings are used due to their effectiveness. Word embeddings are a type of word representation that allows words with similar meaning to have a similar representation (Brownlee, 2019). They are a class of techniques that map words to real-valued vectors in a predefined vector space (Brownlee, 2019). Each word is a set of numbers that together fill the vector space. The representation of the words is also affected by their meaning, allowing for words that are used in similar ways to have a similar vector representation. Therefore, the similarity between two word embeddings will allow us to find their level of similarity, for example, the word embeddings for king-queen and man-woman will be kept similar (Costa, 2019).

With the advances in deep learning and the increasing availability of text corpora for the production of word embedding models, some alternatives were developed (Costa, 2019). These include word2vec (Word2Vec), Glove (Pennington et al., 2014), ELMo (Alammar), ULMFiT (Howard and Ruder, 2018) and BERT (Devlin et al., 2018).

2.6 Applications of NLP

There are several tools and applications used daily by people, that utilize NLP, to process what they said or wrote and execute an action they require. These applications consist of, for example, speech recognition, sentiment analysis, question answering, automatic summarization, chatbots, text classification, and spell checking (Khurana et al., 2017). The following applications are the ones most relevant to this dissertation:

2.6.1 Speech Recognition and Text Classification

NLP advances have allowed the area of speech recognition to achieve significant success. This method of technology is being used to replace other methods of input like typing, clicking, or selecting text (Kharkovyna, 2019).

As mentioned in section 2.2, this is used in existing voice assistants like Amazon Alexa, Siri, and Google Assistant.

Text classification is the process of labeling pieces of text according to its content (MonkeyLearn, 2020). Text can be extremely rich in information, but extracting valuable information from unstructured data can be a hard and time-consuming process (MonkeyLearn, 2020). Businesses are turning to text classification, or text categorization, to organize, structure, and categorize

any type of documents they want (MonkeyLearn, 2020). To perform this classification task, two approaches are available, rule-based and machine learning-based.

2.6.1.1 Rule-Based

The rule-Based approach involves manually generating rules that instruct systems to use semantically relevant elements of a text to identify relevant categories based on its content. For example, in a news article, the goal is to determine its category between politics or sports. Firstly, two lists of words must be defined that characterize each group, for example, "football, basketball" for sports and "Donald Trump, Hillary Clinton" for politics. Then to classify an incoming text, the approach is to simply count the number of words that appear in a document pertaining to politics and the number of words that belong to the sports group. If the politics word count in the article is higher than the sports word count, the article is classified as a politics article and vice-versa (MonkeyLearn, 2020). A rule-based classifier is a technique for classifying records using a collection of ”if ... then ...” rules (Qin et al., 2009).

This approach is human friendly since it can be easily understood, however, it requires a deep knowledge of the domain that is being worked on (Qin et al., 2009). Furthermore, building a good performing rule-based system is time-consuming and requires a lot of analyzing and testing. These systems are also not sustainable since adding new rules and words is a possibility and these can affect the results of the already integrated rules (MonkeyLearn, 2020).

2.6.1.2 Machine Learning

Using machine learning over manually crafting classification rules has the advantage of being able to learn to make classifications based on past observations. That learning method uses pre-labeled text as training data, which teaches the machine learning algorithm to learn the correct associations between pieces of text and their respective labels (MonkeyLearn, 2020; Ikonomakis et al., 2005).

Training a classifier starts with feature extraction, which transforms every piece of text that it receives into a vector of numbers. One model that is commonly used to perform the previous task is "Bag-of-Words", where every produced vector of numbers represents the frequency of a word in a predefined dictionary of words (MonkeyLearn, 2020; Sheikh et al., 2016).

In figure 2.3 it is displayed how the training of a classifier model begins and ends. First, there is the need for datasets of labeled text which will be processed in the feature extraction phase and transformed into a numerical representation. With the features extracted and the tags specified, the machine learning algorithm will be trained and in the end, a trained classifier model is built ready to receive text and produce predictions. In figure 2.4 the process is similar in the way that text still is used as the input and it still is processed in the feature extraction phase, however, since the goal the classifier model was previously trained to classify unlabeled information it now can take the text from the input and attribute a tag to parts of the text as it sees fit based on the previous training.

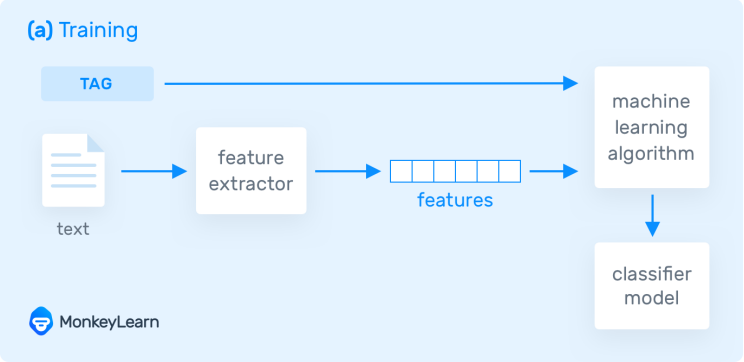

Figure 2.3: Training a classifier model (MonkeyLearn, 2020)

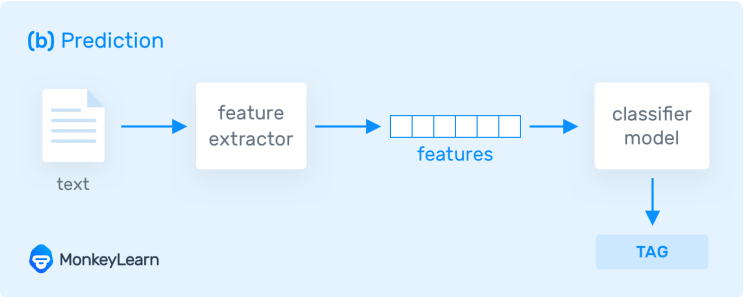

Figure 2.4: Using a classifier model for tag prediction(MonkeyLearn, 2020)

Using machine learning over human-crafted rules is very advantageous since the trained model is usually more accurate and doesn't require the human resources needed for manual classification. The trained model can also be further trained to perform predictions on new examples, being easier to maintain.

In the machine learning approach, there are also varied algorithms that can be utilized to train a classifier model such as Naive Bayes, Support Vector Machines, and Deep Learning models that will be explored in the next chapter.

- **Naive Bayes**

  A Naive Bayes classifier is a probabilistic machine learning model that's used for classification tasks. The crux of the classifier is based on the Bayes theorem (Gandhi, 2018). Naive Bayes can be used for document classification or spam filtering, which have produced great results even with small training datasets (scikit-learn developers).

  Naive Bayes classifiers also take advantage of their fast speed when compared to other available algorithms as seen in the scikit-learn page (scikit-learn developers): "The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one-dimensional distribution. This, in turn, helps to alleviate problems stemming from the curse of dimensionality".

- **Support Vector Machines**
  Support Vector Machines is another algorithm for text classification which shares the same benefit from Naive Bayes of not requiring high amounts of training data. However, it can produce more accurate results by needing more computational resources. In short, SVM takes a space that contains vectors that belong to one group and another space that contains vectors that belong to another group and draws a line between them so both of them are separated from each other. These vector representations represent text that is labeled with a category, therefore, the pieces of text are being separated by category (MonkeyLearn, 2020; Sweilam et al., 2010).

2.7 Conclusion

This chapter explores the definition and uses of health records and electronic health records by analyzing the specifics of medical notes, the typically used note format(SOAP), and the different modes for clinical documentation as well as their challenges. Speech recognition is also detailed and focuses on the current state-of-the-art systems that transcribe speech into text. Natural language processing is also explored in detail as well as Word Embeddings and the applications that most fit this thesis requirements such as text classification. In the end, it is presented two published works for SOAP classification and how this thesis differentiates itself from their approaches.

Chapter 3

Methods and Tools

3.1 Introduction

This chapter explores the methodology undertaken to ensure the fulfillment of the thesis' research objectives, this includes the tools that were used such as Google Cloud's Speech-to-Text solution for automatic speech recognition and the BERT model responsible for structuring transcriptions. The data that was used to train the model, along with its quantity is also defined. Therefore, the methodology is divided into two phases, transcription of spoken dialogue and automatic generation of clinical notes.

3.2 Transcription of Physician-Patient dialogue

In order to get the best audio capture, there was a need to study the available speech-to-text solutions and determine which one most accurately transcribed dialogue into text, taking into account the necessity of European Portuguese support as discussed in the previous section 2.4.1. However as will be discussed in detail in the chapter 4, due to a lack of data in European Portuguese, the solution was adapted to work with the English language.

The accuracy of the speech-to-text system was a major priority in the process of choosing the best solution. The Google Cloud's Speech-to-Text API proving to be the best was chosen as the lead candidate for testing. The API was also selected due to the fact that it can be easily used in a mobile device, which is a main focus of this dissertation, and also because Google Cloud provides sample apps on their website (GoogleCloudPlatform).

Since the calls to the API are not unlimited, shorts tests of Portuguese Medicines' names were also done to test the ability of the API to recognize medical terms. From a set of 24 medicine names from the Cabo Verde list of national medicines (WHO) and Infarmed (Infarmed) list of nonprescription medicines, the API successfully transcribed, from speech to text, 22 of them. "Benzilpenicilina", "Bisoprolol" despite being slightly hard to pronounce were accurately transcribed but "Cadeína" and "Ibuprofeno" failed to get the correct transcription.

3.2.1 Improving the chosen solution

Despite being the most accurate solution, Google Cloud's Speech-to-Text API still has one major drawback, which is the lack of punctuation in the transcriptions. As previously mentioned, a main objective is to be able to generate a clinical note by categorizing sentences, based on the SOAP categories. Therefore, it becomes essential to have punctuation in the transcriptions so that all the sentences can be extracted and properly labeled.

The python package found in (Spencer) was chosen to perform automatic punctuation. It consists of a bidirectional recurrent neural network model with an attention mechanism for restoring missing inter-word punctuation in unsegmented text. The model was trained using 2,121,889 sentences(52,300,149 words), in European Portuguese, from the European Parliament Proceedings Parallel Corpus found at (Koehn, 2004), achieving good results for the punctuation "Period" as seen in the table 3.1.

With this punctuator model available, one of the drawbacks presented before is made less severe and now allows for the extraction of sentences based on "Period" used in the second phase.

Punctuation	Precision	Recall	F-Score
Comma	75.0	72.6	73.8
Period	82.3	81.9	82.1
Question Mark	71.4	51.7	59.9
Exclamation Mark	20.0	0.0	0.1
Colon	60.6	29.3	39.5
Semicolon	0.0	0.0	0.0
Dash	53.0	5.6	10.1

Table 3.1: Trained punctuator results

3.2.2 Mobile Development Frameworks

Choosing the right framework for mobile development is often difficult as there are plenty of existing frameworks, each with different advantages and disadvantages. The process starts by deciding which of the following types of frameworks, Hybrid, Cross-Platform, or Native, best applies to one's needs(SPAssurance, 2018):

The hybrid approach allows developers to create a single mobile app using standard web technologies, such as HTML, CSS, and Javascript, that can be distributed across multiple operating systems at once with minimal to no changes(Huynh et al., 2017; SPAssurance, 2018).

A Cross-Platform development framework offers the same approach as the hybrid approach. However, it replaces HTML and CSS with languages like Javascript, Ruby, or Java, and the code is processed by a cross-compiler that transforms it into platform-specific native apps(TRIBAL, 2011; SPAssurance, 2018). This approach results in better performance and improved user experience since the apps behave like a regular app on the user's ecosystem (TRIBAL, 2011; SPAssurance, 2018).

The top 3 cross-platform frameworks are React Native, Angular 2 Nativescript, and Xamarin (SPAssurance, 2018). Each with its own set of features, but since most features won't be needed, there is no need to further analyze each framework.

React Native has gained a vast and mature community since it was first launched in March of 2015(Bartosz Skuza, Agnieszka Mroczkowska, Damian Włodarczyk, 2019). Its technology is easy to learn and there are plenty of tutorials and libraries, to fasten development(Bartosz Skuza, Agnieszka Mroczkowska, Damian Włodarczyk, 2019).

The Cross-Platform Native frameworks are superior to the Hybrid framework in the sense that the resulting app performs better, follows the user's platform ecosystem(TRIBAL, 2011), and adds no extra development costs (Garbade, 2018b).

Native development involves using each platform's language, tools, and frameworks, such as Apple's and Google's tools, to take advantage of specific operating system features and installed software programs(SPAssurance, 2018; Praveen Kumar S, 2014). As previously mentioned, Google Cloud provides the source code for a basic Android app that can already perform speech recognition in English (GoogleCloudPlatform). To save time, and guarantee secure connections between the phone and the Google Cloud's services, the available source code was modified for the needs of this dissertation, including implementing new features and changing the design.

3.3 Generating clinical notes from Physician-Patient dialogue transcriptions

In this phase, the main objective is to be able to process the text resulting from the spoken conversation between a physician and a patient transcribed into text and produce a clinical note. The SOAP format discussed previously allows for the creation of a sentence classifier that takes as input the sentences from the previous transcriptions and labels them between the four categories. These categories are subjective, objective, assessment and plan. The subjective category mostly contains the experiences and complaints that are said by the patients. The objective category contains factual data collected by the physicians such as vital signs, weight or temperature. The assessment category contains a diagnosis which takes into account the collected subjective and objective information. The plan category contains the steps that will be taken for the future care of the patient. In order to train the sentence classifier model, a dataset of labeled training samples was needed.

3.3.1 Dataset

A reoccurring problem when dealing with data based on medical information is the small amount of it publicly available as well as the possible breaches of privacy since medical data often contains patient information. Since the data used in hospitals is of real patients it can't really be accessed

online. To overcome this problem, some samples that are created by real physicians based on real interactions, while erasing any information that could identify a real patient, are available online.

Fortunately, there is one recent English dataset, of COVID-19 and pneumonia consultations, that contains medical dialogue between patients and doctors, with the doctors providing advice (Ju et al., 2020).This dataset consists of 603 written consultations resulting in 4,184 unlabelled sentences with an average of 111 words per consultation. 166 medical notes in the SOAP format, which were created as samples and do not contain any private information, were also collected (MTHelpLine). The dialogue from the consultations which is not spoken but written, does lose several aspects that can be retrieved from a spoken conversation, such as gestures, facial emotions, body language or nods. However, the dataset is recent and is still from dialogues, which remains useful for experimentation. Since data is hard to come by, its useful to study every piece that is found.

One objective of this thesis was to perform the automatic note generation in European Portuguese, however, due to the low amount of publicly available resources, most of the data is in English or other foreign languages. Since the data is not in European Portuguese, all of the relevant English data was collected and aggregated to build a larger dataset to be used for training. Despite being in a different language, the dataset can still be used for model training to analyze if the implemented solution is viable or not. If it is viable, there is a high probability that it will perform adequately in European Portuguese when data in that language is available.

3.3.1.1 Labelling Sentences

To be able to train a sentence classification model there need to be labeled sentences and since the 4.181 sentences available in the dataset do not contain any label, the manual process of labeling sentences must be done. This process was done by me with the help of 2 informatics colleagues who labeled a few hundred sentences which I later reviewed. Since neither of us study in a medical domain, the guidelines for the filling of SOAP notes were followed, proving to be an understandable structure even for people outside the medical domain. Taking into consideration the SOAP format, each sentence must belong to each of its categories or if not belonging to any of them, must be labeled as such. Subjective phrases were labeled as "Subjective", objective as "Objective", assessment as "Assessment", plan as "Plan" and irrelevant sentences as "Irrelevant". To accurately label the sentences some patterns were followed for each of the categories:

1. **Subjective**

   The subjective category is based on the patient's personal view or feelings therefore sentences that contain symptoms provided by the patients or narration of recent events in the past tense is often a subjective sentence.

Patterns	Description
"I have <symptom/disease>", "I don't have <symptom/disease>", "I feel <symptom>"	This one is very common, the patients start the conversation by saying they have some symptom or disease, for example, "I have a cough".
"I took < X medicine > for < X amount of days >", "they worked", "they did not work"	When a patient talks about their medication and effects.
"I was in contact with someone that later tested positive for Covid-19"	It is common for patients to point out that they have been in contact with someone that was later tested positive for Covid-19 or presents the symptoms related to Covid-19.

Table 3.2: Subjective decisive patterns

2. **Objective**

   Since this dataset was obtained from a dialogue between a patient and a doctor over the internet in written format, the objective data that is gathered by being in contact with the patient and measuring, for example, his vital signs and physical exams is very small and insignificant. To combat this, a set of 166 sample clinical notes in the SOAP format from the website mtsamples.com were used to retrieve the objective content and balance the dataset.

Patterns	Description
"Vital signs are ...", "Weight is ...", "Heart bate is ...", "Blood pressure is ..."	These are the most common. The physician writes in the clinical notes the measurements he takes from the patient.
<EXAM NAME> reveals ...	The findings of exams determined by the physicians.

Table 3.3: Objective decisive patterns

3. **Assessment**

   Sentences that describe the assessment of the physician on the patient. These sentences usually have the word ‘diagnosis’ in them along with a disease or medical condition. The number of sentences belonging to this category was also very little so to combat this and balance the dataset, the assessment section of the 166 sample clinical notes was extracted.

Patterns	Description
"<Disease Name>", "Diagnose"	A lot of assessment sections simply contain the name of the diseased diagnosed by the physician.
"The issue appears to be ..."	Determining the diagnosis based on the subjective and objective content.

Table 3.4: Assessment decisive patterns

4. **Plan**

   Sentences that describe plans of the physician for the future treatments on the patient. Every step that is being taken for the care of a patient and the need to consult with other clinicians is written in this section.

Patterns	Description
"Avoid contact with", "test for Covid-19 when possible", "stay at home"	These are recommendations stated by physicians that apply to the current Covid-19 pandemic that should keep the patients safe.
"Take antibiotics", "Take anti-inflammatory medicines"	Medication recommended by the physicians.
"You should do <X exam>", "Go to your local hospital and confirm your <Disease/Symptom> "	Physicians will often recommend the patients to perform exams, or contact with other clinicians, to better assess their current condition.

Table 3.5: Plan decisive patterns

5. **Irrelevant**

   Every sentence that does not belong to any of the previous categories was labeled as ’irrelevant". Most of these include questions asked by the patients. This is also important since, in a dialogue, a lot of information will be transcribed but not all of it will be relevant and belong to one of the SOAP categories. These also need to be identified and categorized in their section to allow for the remaining sentences to more accurately be categorized to generate the best possible clinical note.

After labeling the dataset and gathering the publicly available data, our current updated dataset contains a total of 8.097 labeled sentences with the distribution presented in the table 3.6. This data is now usable to train a model in a text classification situation.

Category	Count
Subjective	1.380
Objective	1.422
Assessment	2.194
Plan	765
Irrelevant	2.336
Total	8.097

Table 3.6: Number of labeled sentences by category

3.3.2 Using BERT for Sentence Classification

With the training data collected and each sentence labeled with a specific class from the SOAP medical note structure or marked as irrelevant, it is now ready to be used in a sentence classification problem. The problem in this situation is being able to categorize new incoming sentences without any labels and attribute the correct category to each of the sentences.

Since 8097 of sentences is not a very high amount of sentences, taking into account that at least 15% will be used for testing the model and the rest for training it, there was a need to explore the currently available state-of-the-art solutions that could handle a small dataset and still produce adequate results. BERT was found to be a great solution to be explored.

The BERT (Bidirectional Encoder Representations from Transformers) model that was introduced in the paper BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (Devlin et al., 2018), made achieving state-of-the-art results in a variety of NLP tasks possible, while also not requiring a large dataset and being open source.

BERT takes advantage of the Transformer attention model and applies bidirectional training which takes into account the previous and next elements of a text, to better assess the context. This provides better results when compared to other methods that analyze a text in one direction, from left-to-right or vice-versa (Horev, 2018). An example of how the Bidirectional training method works is displayed in figure 3.1.

As previously mentioned, BERT takes advantage of the existing Transformer attention model to learn context between words in a text. The transformer model makes use of two separate mechanisms, one that collects text and another one which produces predictions based on the collected text. In the case of BERT the only mechanism that is useful is the second one since BERT's objective is to generate a language model (Horev, 2018).

The most important part about BERT is that it is trained on a large text corpus, containing 800 million words from the Book Corpus and 2,500 million words from Wikipedia (Zhu et al., 2015). By being trained with such a large number of words, BERT starts to better understand the nuances and context of how language works (computer science graduate, 2020a). To better

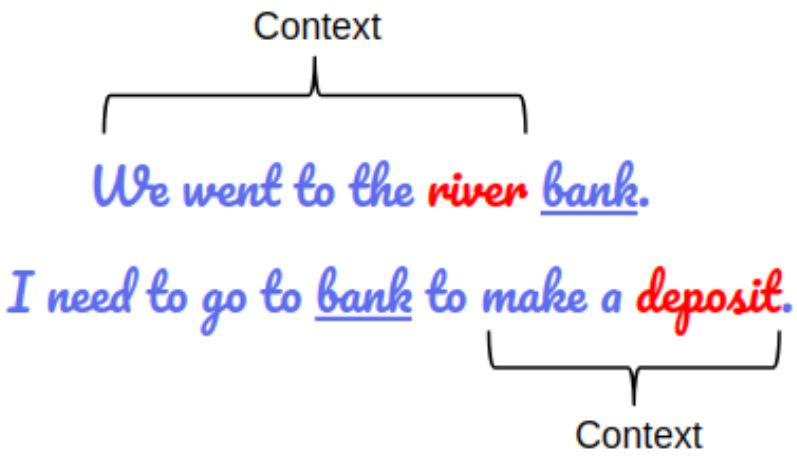

Figure 3.1: Bert takes into account the left and right context (Horev, 2018)

understand how BERT fits into the situation it is better to explore how the quest for learning language representations started, adapted from (computer science graduate, 2020b):

3.3.2.1 Word2Vec and Glove

The quest for learning language representations by pre-training models on large unlabelled text data started from word embeddings such as Word2Vec(Word2Vec) and GloVe(Pennington et al., 2014).

The previous word embeddings contain additional information that makes the development of NLP tasks provide better results even without the use of large amounts of task-specific data. However, one major downside of using this approach is that the context of the analyzed text is not taken into account, which can provide the same result for words that are written the same, but in their context have entirely different meanings. Also, due to the use of very shallow Language Models, more complex language models such as layers of Long short-term memory networks (LSTMs) and Gated Recurrent Unit (Grus) started being explored to fix the limited amount of information the shallow language models could capture.

3.3.2.2 ELMo and ULMFiT

As evidenced in the previous two word embeddings, one key downside from using them is that two words that are written the same but have different meanings in their context will produce the same result. ELMo (Peters et al., 2018) was created to target that exact problem and it does so by not using shallow language models but by using layers of complex Bi-directional LSTM architectures. Therefore, two words that are written the same but have different meanings will produce different ELMo embeddings.

The following effort was the creation of the Universal Language Model Fine Tuning (ULMFiT) (Howard and Ruder, 2018) that could be used to fine-tune trained models even with small amounts of data, capable of performing classification tasks. Using previously-stored knowledge to solve a problem could then be further enhanced to apply to another problem and provide great results, this became known as transfer learning which in this context is equal to pre-training plus fine-tuning a model.

3.3.2.3 BERT

Reaching state-of-the-art performance was and still is one of the major objectives that researchers try to reach. Therefore, with continuous research, BERT was created. Despite ELMo providing great results, there are some differences between it and BERT which must be taken into account, the following are based on (Alammar):

1. Truly Bidirectional
   Due to its use of masked language modeling techniques, BERT becomes a truly bidirectional model. In contrast with the other available models such as ELMo which uses concatenated left-to-right and right-to-left LSTMs and ULMFiT which uses a unidirectional LSTM.

2. Model Input
   BERT tokenizes words into sub-words (using WordPiece) and those are then given as input to the model. ELMo uses character-based input and ULMFit is word-based. It has been claimed that character-level language models do not perform as well as word-based ones but word-based models have the issue of out-of-vocabulary words. BERT's sub-words approach enjoys the best of both worlds.

3. Transformer vs. LSTM
   While ELMo and ULMFit use LSTMs for training, which can become a slow process the higher the amount of data. BERT uses transformers which provide a way to parallelize the training process and speed it up by partitioning a model in different parts and using them on the different available resources such as the graphics card.

Chapter 4

Implemented pipeline for the automatic creation of medical notes

With the improvements done to speech recognition technologies and more specifically the continuous increase in the accuracy of Portuguese transcriptions from speech to text, it becomes a possibility to use these advances to ease the burden put on physicians when they have to write medical notes. To reduce the amount of time spent writing on a keyboard, but still, keep the same standards and follow the correct guidelines to produce medical notes, is what a speech to text system can help achieve. In addition to the previous systems, new NLP techniques that allow for the automatic extraction or classification of unstructured text also can, if used in the medical context, and in conjunction with the speech to text systems, automate the analysis or structuring of medical notes.

As mentioned previously, the main objective of this thesis is the development of a fully automated pipeline capable of structuring transcribed dialogue from a physician-patient appointment in order to create a medical note that will be inserted into the EHR. This in turn would, in theory, help alleviate the amount of documentation work done by the physicians as well as increase the amount of time the physicians spend looking at the patient while reducing the time spent writing and looking at the computer. This chapter thoroughly details and explains every process that occurs in the pipeline, from the transcription of text to the creation of a medical note, along with their unique setbacks and breakthroughs.

The architecture of the proposed pipeline can be seen in Figure 4.1, delineating every major step that is done to ensure that the previous objective is accomplished as well as the means to achieve success in those steps.

4.1 Speech-to-Text

Transcribing the dialogue from a conversation between a patient and a physician into text, is a crucial initial step, since it is the basis from which every following process will act upon. As discussed in the Section 2.4, the Google Cloud Speech-to-Text solution was chosen to transcribe

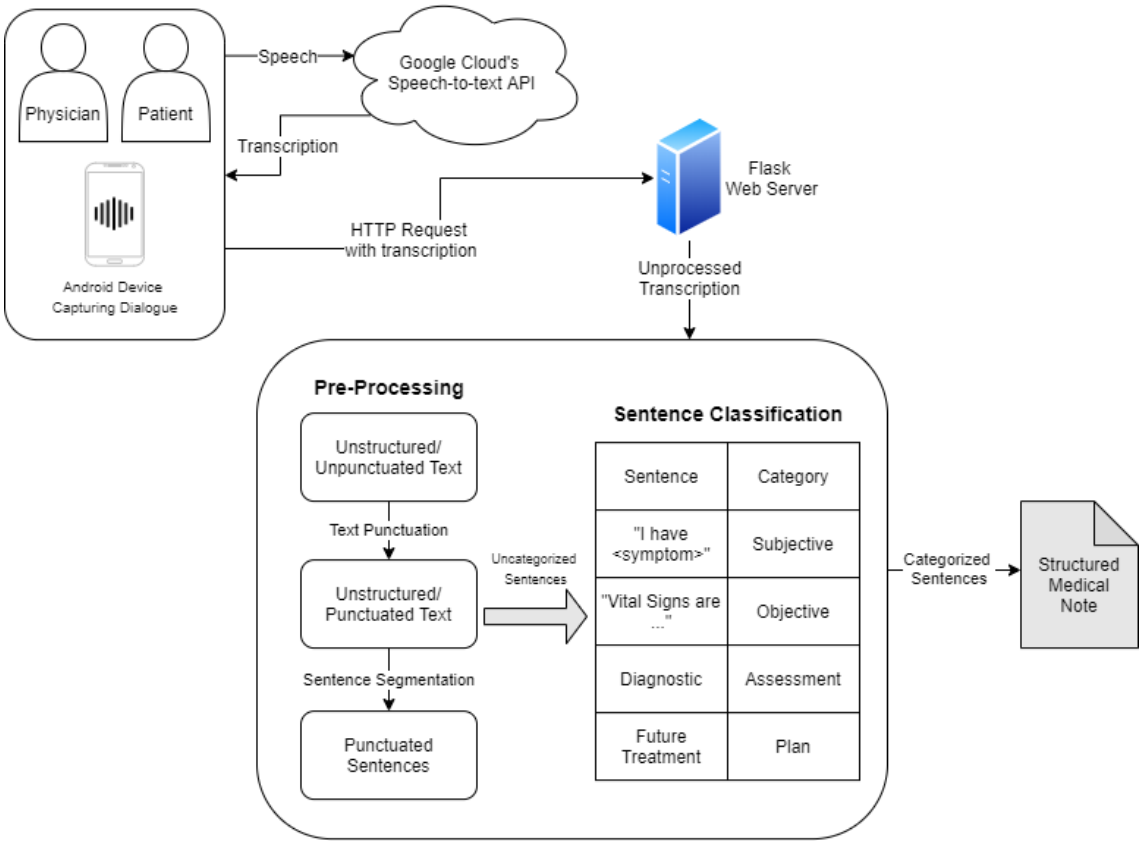

Figure 4.1: Architecture of the pipeline responsible for collecting voice from a dialogue between a physician and a patient using an android application and creation a medical note using sentence classification with the unstructured, unpunctuated transcribed text.

the dialogue due to its support of the European Portuguese Language as well as being the most accurate system from the list of proposed speech-to-text solutions. In order to use the Google Cloud services, a free one-year trial provided by them was used, being enough time to use their technologies for the development of the pipeline as well as testing it. In addition to the free one-year trial, Google also offers 300$ alongside the 60 free minutes per month.

One of key features of the proposed pipeline is to be able to reduce the time spent on documentation by the physicians, therefore to best accomplish this task, a mobile application with a near real-time speech to text transcription was chosen as the optimal solution. This way, the physician can easily check if the transcription is being done as well as detect any missing sentence that wasn't caught by the application.

4.1.1 Mobile App

Following the assumption that near all physicians possesses some type of mobile device, and consequent proof (Mobasheri et al., 2015), it was decided that in order to avoid adding another external device to the architecture, like a microphone, a mobile app would be best suited for any task pertaining speech recognition. Therefore, to capture the dialogue between a physician and a

patient during an appointment, an android application was developed and tested in a Xiaomi A1 android device. The user interface for the transcription consists of one screen presented in Figure 4.2 that will contain all current transcriptions retrieved from Google Cloud, before the user decides whether he wants to save them or delete them.

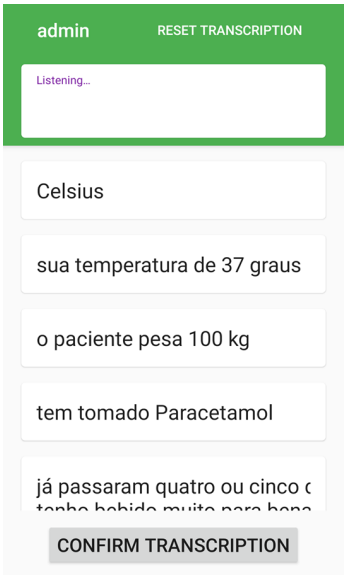

Figure 4.2: Transcription of speech to text in European Portuguese

The android application is based on a sample application from the Google Cloud platform that uses Streaming recognition, which is near real-time transcribing, to provide direct transcriptions to the user. As detailed in Google Cloud's documentation: "Streaming Recognition performs recognition on audio data provided within a gRPC bi-directional stream." (Google, 2020). gRPC is a modern lightweight communication protocol from Google that offers better performance and security compared to REST+JSON (MSV, 2019). The streaming recognition is most useful when there is a need to immediately assess the results from the transcriptions, while a person is still talking or immediately after the person stops talking. This is where the gRPC communication protocol displays its main feature, by making it possible to simultaneously send speech to the Google Cloud services and retrieve responses during a dialogue (Google, 2020).

An ideal scenario for the use of the app would be following. The physician lays his phone on a table, between him and the patient, opens the app, and it automatically starts transcribing dialogue. The streaming recognition is used to avoid storing audio clips on the phone that would need to be sent to the Google Cloud's servers for transcription. The larger the audio clip, the longer it would take to upload it to the Google Cloud's servers. Since the streaming recognition is in near real time, it is the best option for this scenario.

The main goal with the app is to provide a simple and unobtrusive way for a physician to get a transcription from the Google services. However, to access the transcriptions he has performed, a server and a website were created to allow him to send transcriptions to his account and retrieve them from the website. These will be further explain in the following sections.

4.2 Connecting physicians to transcriptions

When a physician first starts an appointment he can open the previously mentioned mobile app and start transcribing speech into text. However, since the goal is for the resulting text to be processed before it is inserted into an EHR, that text needs to be transferred from the smartphone to the computer, capable of performing more demanding NLP techniques, so that the physician can edit as he sees fit and copy it to the EHR. Therefore, a website running on a flask based server using a PostgreSQL database for the storage of transcriptions and registered users was built alongside the development of the app.

4.2.1 Implemented Server and Database

As previously mentioned, the implemented server was built using the Flask web framework and it connects to a PostgreSQL database. How these work and how they were used in this dissertation is explained with the following:

What is a web framework?

"A Web Application Framework or a simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management, and so on." (pythonbasics)

What is Flask?

"Flask is a web application framework written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine. Both are Pocco projects." (pythonbasics)

Why is Flask a good web framework choice?

It is easy to get started with and can produce a working web application with only a few lines of code. It also is very pythonic which allows for the usage of NLP technologies available with Python (pythonbasics).

The server uses a PostgreSQL database, which is compatible with Flask, to store the profile of the users that will authenticate into the system. It is also used to store all of the transcription sent to the server by the users. PostgreSQL was chosen due to it being able to handle multiple users at the same time as well as being secure. The diagram 4.3 details the tables defined in the server's database.

4.2.2 Implemented Mobile Application

As detailed in section 4.1.1 a mobile app was developed to allow for the transcription of speech in near real-time during appointments. However, these transcriptions need to be sent to secure remote storage so that they can be processed and accessible by each user. Therefore, a login system was created to protect data between users.

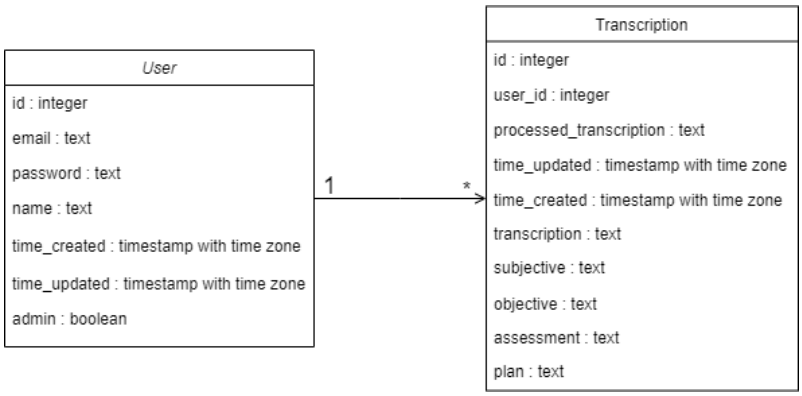

Figure 4.3: Database Tables

4.2.2.1 Authentication

The flask-JWT framework was implemented to allow for the android device to authenticate to the server and keep a secure connection. As seen in figure 4.4 the user will enter his authentication details that will be sent over to the developed server with a POST request in JSON format. If a user exists with those credentials, the server will send back a positive response containing a token that will validate future transactions with the server. This token is called access token and will be used to verify that the user is logged in to the server with a valid profile so that he can perform authenticated features.

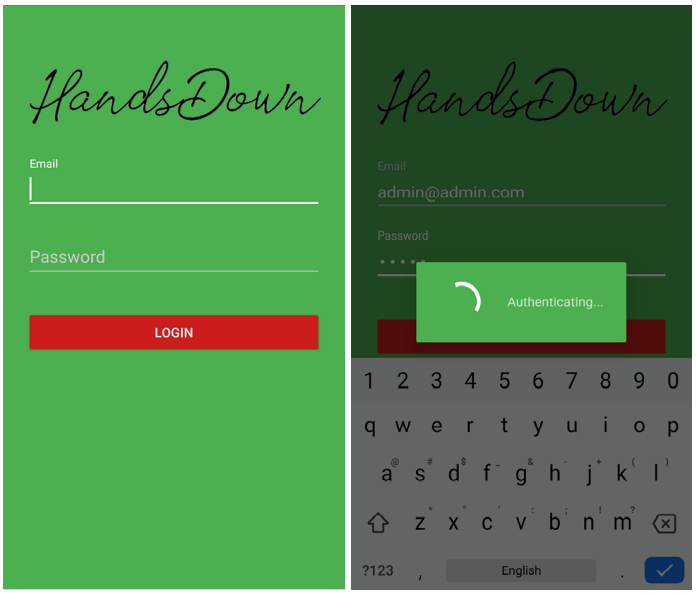

Figure 4.4: Authentication in the android application

4.2.2.2 Saving transcriptions

After successfully authenticating himself and performing the desired transcription tasks, the user will want to save those transcriptions to his account so that he can access them later. As seen in

figure 4.5 when the user desires to save the collected transcriptions and store them in his account he presses the button to send transcription which takes him to a screen where he can review the transcription that will be saved. When the user confirms the transcription it will be sent to the server where it will be saved, processed, and saved again, so that he can always check the full transcription to compare to the structured one.

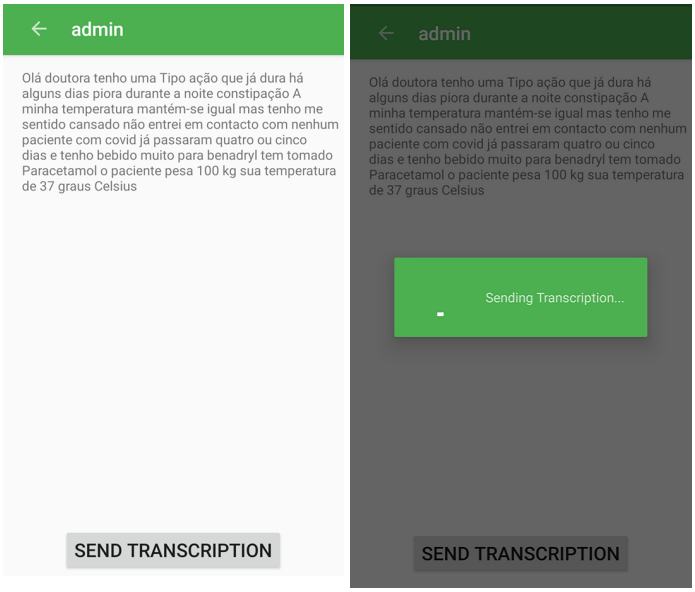

Figure 4.5: Saving transcription

4.2.3 Implemented Website

In order to provide users with the means of accessing all of their previously collected transcriptions in a simple and interactive way, a website was chosen as the best solution. The website is run using a python module named Flask which is a web framework capable of quickly producing a web application.

When the user opens the website he will be treated to a welcome page with a top section containing the actions he can perform. If it is the first time being accessed the user won't have access to any authorized features since he is not logged in.

The authentication is performed using Flask-Login which takes an email and a password as input to verify whether or not a profile with those credentials exists in the database. Flask-Login is easy to be integrated and handles the required authentication tasks such as logging in, remembering sessions, and logging out (auth0). It also blocks unauthorized users from accessing authorized pages.

When authenticated the user can logout from the website by simply clicking the logout button present on the top section.

After login, the user will see a list of all the transcriptions he had previously sent from the mobile application to the server, ordered by date, that are currently associated with his profile as

seen in Figure 4.6. He can browse through all of them and click each one to better examine each transcription.

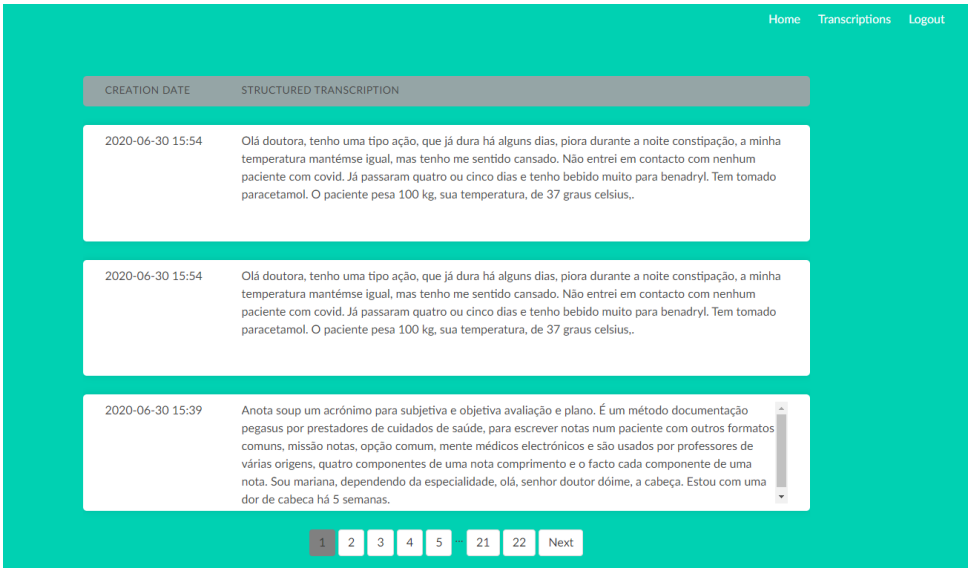

Figure 4.6: List of transcription

By clicking a transcription from the list of done transcriptions the website will take the user to a page containing the chosen transcription where he can analyze the fully untouched transcription collected from the mobile application, on the left side, and compare it to the structured transcription on the right side. The user can choose to edit the structured transcription to update its contents to better suit his needs. This is displayed in figure 4.7.

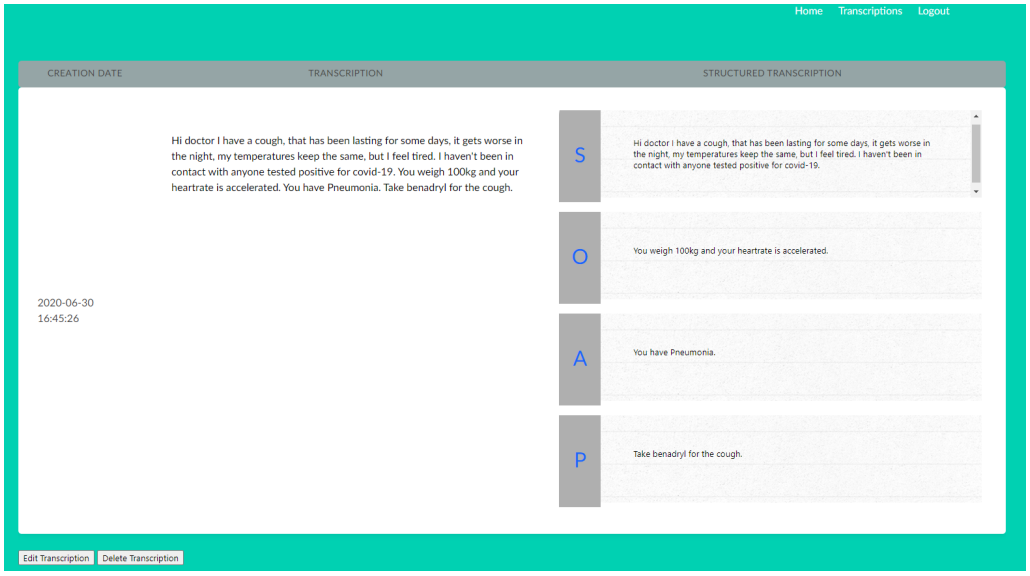

Figure 4.7: Transcription

4.3 Natural Language Processing in Transcriptions

As described in the previous section there are two very important elements in the system, one being the phone capable of using the microphone to capture speech and transcribe it into text and the other being the website where the users can inspect their transcriptions. Connecting and supporting both of them is the Flask server which is developed using Python, which handles the authentication and storage of transcriptions. However, since the main objective of this thesis is to produce a structured medical note, when the user sends the transcription from the mobile application to the server, the server needs to process the transcription before storing it. This is where the Flask web framework is useful since it is based on python, we can use the python natural language processing libraries and modules to process text while also working as a server connected to a database.

In section 3.3 a solution was explored using BERT that receives sentences as input and attributes a category from the SOAP format to each of them. The following steps explore in-depth the proposed solution, involving training a BERT model for sentence classification. However, taking into consideration that the transcriptions contain no punctuation and are not separated into sentences, the steps required to prepare the transcriptions for classification will be detailed.

4.3.1 Fine-tuning a BERT model for sentence classification

Before the classification of transcription sentences can be possible, the BERT model needs to be fine-tuned to able to understand the SOAP categories and correctly categorize incoming sentences. The following sections will demonstrate the process undertook to fine-tune the model and were based on an existing intent recognition problem (Valkov, 2020).

In machine learning, when dealing with large datasets for simulations, there needs to be processing power to accomplish those simulations and so to facilitate this, Google Colab (Google) was created. Google Colab is an easy to use free cloud service where high computational power is available in a notebook that can be created.

Google Colab contains from the start, support for python 2 and python 3, free GPU acceleration, useful machine learning libraries like TensorFlow, Scikit-learn, Matplotlib among others and each Colab Notebook can be stored in a user's google drive.

A google colab notebook was created to fulfill the fine-tuning process.

The dataset that will be used consists of 1.380 subjective sentences, 1.422 objective sentences, 2.194 assessment sentences, 765 plan sentences, and 2.336 irrelevant sentences for a total of 8.097 sentences as discussed in section 3.3.1.1. All of the sentences were gathered and stored in a CSV file.

The dataset was then split into two files, one for training the model and the other one for testing it after the training is complete. An 80/20% approach was chosen, leaving 6397 training sentences for training and 1646 sentences for testing. The balance of the dataset is shown in figure 4.8.

BERT is open-source and so are the pre-trained models made available. The one that is used here is the BERT-Large, Uncased (Original).

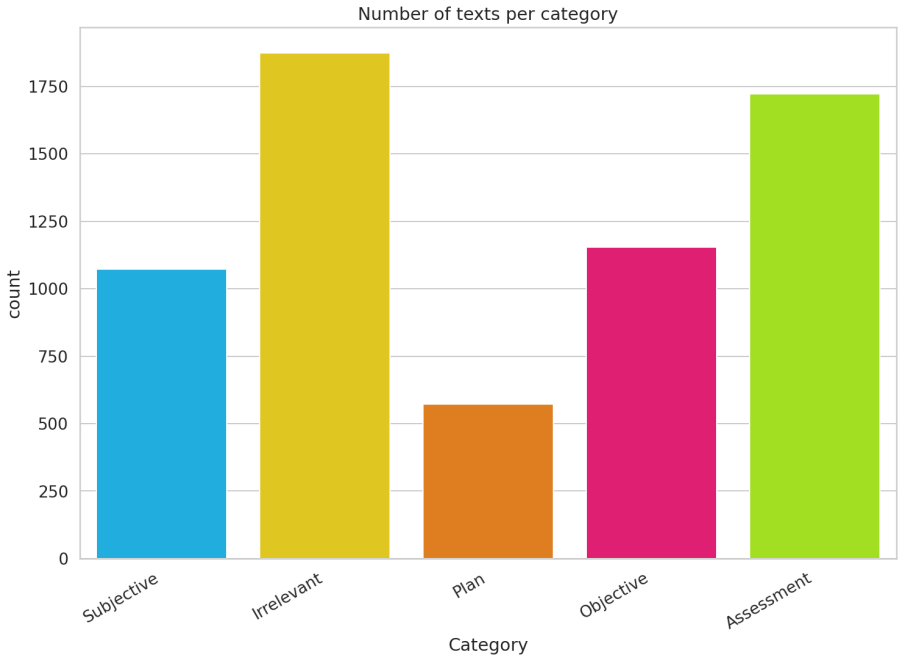

Figure 4.8: Balance of training data

The raw sentences need to be converted into vectors so that they can be used in the model. So 3 steps are performed: tokenizing the sentences, which is the process of separating every word in a sentence, converting the sequence of tokens into numbers, and then pad the sentences so that each one has the same length.

The pre-trained BERT model is fine-tuned, using the collected dataset as inputs (sentence and category). The output is flattened, Dropout with two Fully-Connected Layers is added. The number of outputs is equal to the number of categories there are - 5.

4.3.2 Pre-processing the transcriptions and Categorizing Sentences

All of the transcriptions that are sent by the user to the server arrive without any punctuation. Therefore this is where the automatic punctuator model presented in section 3.3 is useful. The model collects the entire text from the transcription and adds punctuation to it, resulting in punctuated transcriptions.

The second step is to separate each sentence from the punctuated transcription and to perform this sentence segmentation, the Python Natural Language Toolkit is used. The toolkit contains a lot of useful NLP techniques however only the sentence tokenization method will be required, which performs the task of separating a text into sentences based on the "Period".

The third step is to add "[CLS]" before each sentence and "[SEP]" after each sentence. This process is needed for the BERT model to recognize the sentences.

With the previous steps concluded, the process of using BERT to categorize sentences is now possible. Figure 4.9 displays an overview of what was done and the result.

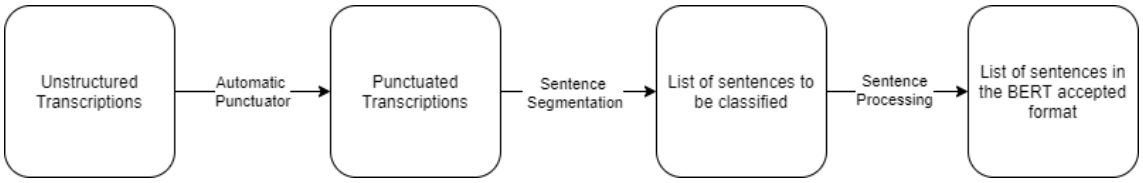

Figure 4.9: Pre-processing text for classification

Having processed the transcriptions and making the text ready for classification the sentences can now be classified and grouped by category as seen in figure 4.10.

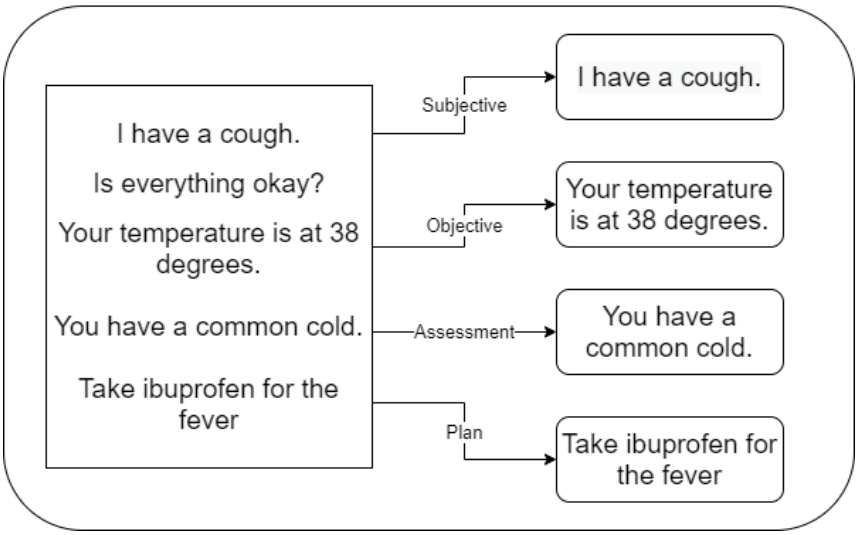

Figure 4.10: Categorizing sentences for the SOAP format

Chapter 5

Evaluation of the Solution

The solution that was developed and explored in detail in the previous chapter can be evaluated by analyzing how accurate the fine-tuned BERT model classifies test sentences into one of the 4 SOAP categories. This chapter explores the process of evaluating a BERT model and its results when evaluated with a test dataset.

5.1 Data

In chapter 3 it was explained how medical information is hard to come by since the data contained within is often sensitive as it can identify a real person. The collected dataset was divided into two files, 80% for training and 20% for testing. The latter is what will be used to evaluate the accuracy of the fine-tuned BERT model. Table 5.1 displays an overview of the number of sentences of each SOAP category contained in the test file.

Category	Count
Subjective	308
Objective	267
Assessment	469
Plan	141
Irrelevant	461

Table 5.1: Number of sentences by category

5.2 Evaluation

For machine learning, there are three methods commonly used for evaluation, which are Precision, Recall, and F1 score. Each one is described below based on (Shung, 2020):

1. **Precision**
   Precision is related to how accurate the models is calculating, from the positive predictions,

how many of them are really positive.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{5.1}$$

2. **Recall**
   Recall calculates how many of the Actual Positives a model captures through labeling it as Positive (True Positive).

$$Precision = \frac{TruePositive}{TruePositive + FalseNegative} \tag{5.2}$$

3. **F1-Score**
   F1-Score might be a better measure to use if a balance between Precision and Recall is desired, and there is an uneven class distribution. Since our dataset is not balanced, this measure will serve as the best indicator for accuracy.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5.3}$$

5.2.1 Evaluating the model

Using the testing data presented in section 5.1 and the previous evaluation methods to test the performance of the fine-tuned BERT model provided some positive results that are displayed in the table 5.2.

The "Plan" category was the only one that could not reach at least 80% of precision, recall, or f1-score, this can be attributed to the low amount of training data that contained "Plan" sentences. By adding more samples to this category, the model will, in theory, provide more accurate results.

Concerning the other categories, the one that performs the best is the "Objective" category. This can be attributed to the factual nature of the category. Since most sentences that belonged to this category contained examinations or observations done by the physician, such as vital signs or a person's weight which always contains a number and the word "Kg" for kilograms.

The "Assessment" category reaches second place in most accurate, which is due to most of the sentences containing few words and always a diagnosis.

Aside from "Plan", the "Subjective" category is the least accurate. This is because this category's sentences always contain dialogue uttered by the patient, mostly in the first-person, that can be wrongly categorized as "Irrelevant". However, accuracy is still high and can be improved with more training data.

It is worth mentioning the importance of the "Irrelevant" category, which contains every sentence that does not belong to any of the other SOAP categories. Therefore, by having a category that collects every sentence that should not be written into a medical note, results in a better-structured document.

Overall, taking into account every category, and the low amount of data used for training, the model's f1-score accuracy still reaches 85%. This is a positive result and demonstrates the possibility of even higher accuracy values with more training data.

	precision	**recall**	**f1-score**	**support**
Subjective	0.83	0.86	0.85	308
Objective	0.95	0.87	0.91	267
Assessment	0.86	0.91	0.89	469
Plan	0.67	0.68	0.67	141
Irrelevant	0.84	0.81	0.82	461
accuracy			0.85	1646
macro avg	0.83	0.83	0.83	1646
weighted avg	0.85	0.85	0.85	1646

Table 5.2: Results of the model evaluation

5.3 Error Analysis

The confusion matrix for the test dataset is displayed in figure 5.1. The justification for the miss-labelling is summarized bellow:

1. **Subjective**
   The subjective category shows to be most similar to the irrelevant category as the latter was attributed to 24 subjective sentences. This is due to the nature of the subjective dialogue since it comes from the patient's speech and most speech can be irrelevant in a medical environment.

2. **Objective**
   The objective category is most similar to the assessment category. This could be due to the fact that when a doctor examines a patient he can say factual data from, for example, an exam, that can be seen as a diagnostic.

3. **Assessment**
   The assessment category has the same amount of miss classification with the subjective and irrelevant categories. The assessment being confused for subjective could be due to the fact that a patient can say a diagnostic himself, even though what he says can only be considered as subjective. In relation to the irrelevant category, it can be due to the fact that when a patient asks a doctor a question containing an assessment pattern or vice-versa, it is considered irrelevant.

4. **Plan**
   The plan category is most confused with the irrelevant category. This could be due to the

fact that when the physician gives a recommendation to the patient in the future tense it could be irrelevant for the medical domain.

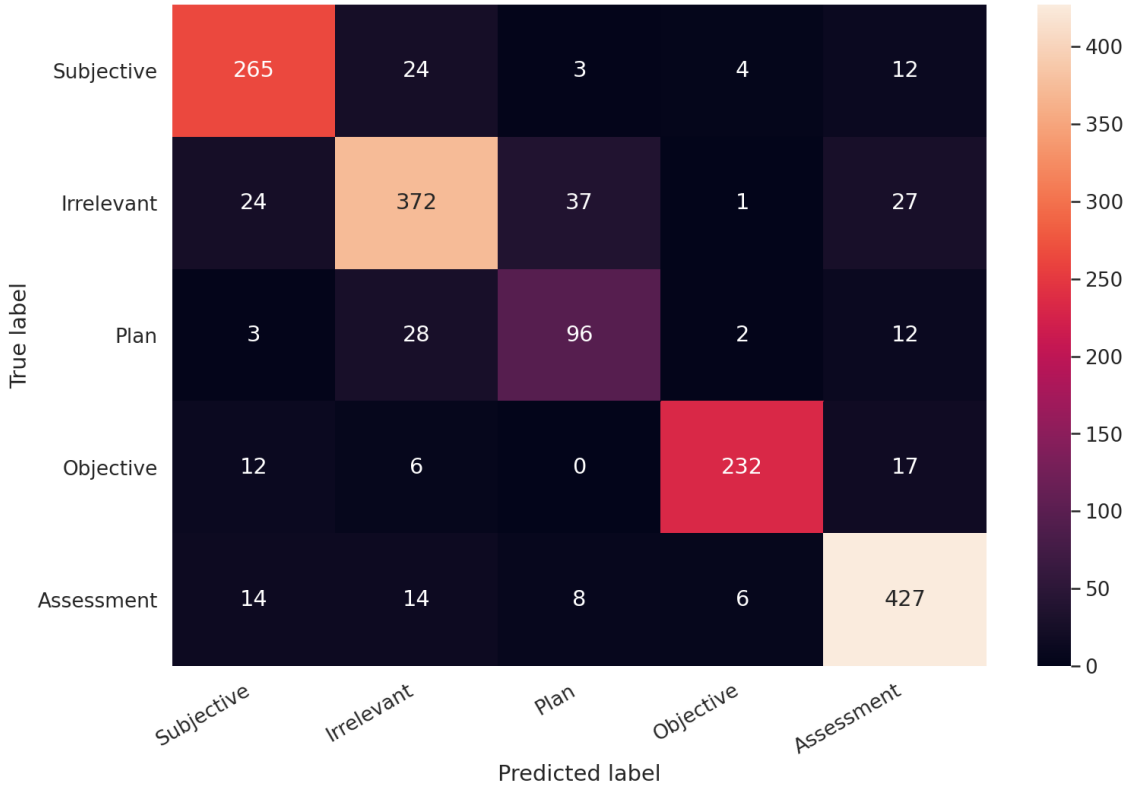

Figure 5.1: Confusion matrix for test dataset

Chapter 6

Conclusions and Future Work

The principal objectives of this dissertation were the evaluation of speech-to-text solutions in a medical environment in European Portuguese as well as the creation of a fully automated pipeline capable of collecting the transcriptions from the speech-to-text solution during a medical appointment, between a physician and a patient, and generating a medical note in the SOAP note structure.

To accomplish the aforementioned objectives a study of the state-of-the-art was done in the fields of speech-to-text and machine learning. Initially, the currently available speech-to-text solutions were investigated and analyzed, these being the Google's Cloud speech-to-text API, Microsoft's Azure speech-to-text, IBM's Watson Speech-to-text, and amazon transcribe. Google's solution proved to be the most accurate in transcribing Portuguese speech into text, according to previous studies, and also the easiest to implement in a mobile environment for android devices. In the field of machine learning, it was explored which technique was best to provide a way to structure the unstructured transcriptions from the physician-patient dialogue. With the discovery of the SOAP medical note structure which is commonly used by medical personnel, it was decided that a sentence classification method to categorize each of the transcriptions' sentences into one of the four categories would be the best approach to evaluate.

The developed solution consists of a system capable of collecting speech, transcribing it into text, sending the text to a server capable of processing it, the server structures the text and then a website is available to the users so that they can inspect their transcriptions in the original format and in the processed format. To collect speech a native android application was developed based on Google Cloud's android sample. The application requires authentication before entering and this authentication is performed by a developed flask web server. After authentication, the user can perform near-real-time transcriptions of speech-to-text in European Portuguese and send it to his profile on the webserver. Before storing the transcription in the PostgreSQL database, the transcription is punctuated, using a deep learning punctuating model, divided into sentences, using the python's NLTK library, and each sentence is classified and grouped together based on their category in the SOAP format, using a fine-tuned BERT model for sentence classification.

Due to the unavailability of training and test data in European Portuguese, the evaluation of the BERT model was performed using an English dataset that consists of English Dialogue

between patients and doctors about Covid-19 and Pneumonia symptoms and assessments. To attempt to balance the dataset is were also added Objective and Assessment sentences, of the SOAP format, from a website of medical samples. The results were shown to be adequate, with near 90% accuracy for the "Subjective", "Objective" and "Assessment" categories, while the "Plan" category almost reached 70%, this means that not enough "Plan" sentences were fed to the BERT model in the training process.

The objectives presented in section 1.3 can now be assessed for their accomplishment status. The intended fully automated pipeline for the transcription of medical appointments in European Portuguese and the creation of medical notes was accomplished with the development of a mobile app and web server to connect speech with natural language processing through easily understood interfaces. This system could not be evaluated in a real medical environment due to the current Covid-19 pandemic and so its performance was only evaluated with sample test data.

6.1 Limitations and future work

Quantity and quality of data in the machine learning domain are what makes the creation of a trained model for a desired NLP task possible or not. The higher the amount of data the more quality data can be extracted. This is one of the limitations in the medical domain since most information is private and sensitive to each patient the amount of publicly available data becomes severely limited. This was noted in the development of the solution. Since the BERT model requires data to be trained and data to be evaluated, the scarcity of training sentences made the model fail to reach 70% in the assignment of the "Plan" category to sentences. Also, due to the current Covid-19 pandemic, the contact between patients and physicians was reduced which made the proposed research question “What impact can the use of Natural Language technology interfaces, that support the filling of EHRs, have on the patient-physician relationship?” impossible to assess.

With the whole system developed and evaluated it was determined that the amount of data collected was the major influence in the quality of the final product. It is predictable that with a larger dataset, especially of physician-patient dialogue in medical appointments, with each sentence manually labeled for the SOAP categories, the BERT-model will more accurately categorize incoming sentences. The system also needs to be tested in a real medical environment in the future.

References

Part of speech (pos) tagging. URL https://www.tutorialspoint.com/natural_language_processing/natural_language_processing_part_of_speech_tagging.htm. Available at https://www.tutorialspoint.com/natural_language_processing/natural_language_processing_part_of_speech_tagging.htm (accessed on February 2020).

Stemming and lemmatization, 2008. URL https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html. Available at https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html (accessed on February 2020).

Erika L. Abramson, Sandra McGinnis, Alison Edwards, Dayna M. Maniccia, Jean Moore, Rainu Kaushal, and with the HITEC investigators. Electronic health record adoption and health information exchange among hospitals in new york state. *Journal of Evaluation in Clinical Practice*, 18(6):1156–1162, 2012. doi: 10.1111/j.1365-2753.2011.01755.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2753.2011.01755.x.

Julia Adler-Milstein, A Jay Holmgren, Peter Kralovec, Chantal Worzala, Talisha Searcy, and Vaishali Patel. Electronic health record adoption in US hospitals: the emergence of a digital "advanced use" divide. *Journal of the American Medical Informatics Association*, 24 (6):1142–1148, 08 2017. ISSN 1067-5027. doi: 10.1093/jamia/ocx080. URL https://doi.org/10.1093/jamia/ocx080.

Sima. Ajami. Use of speech-to-text technology for documentation by healthcare providers. *The National Medical Journal of India*, 29(3):148–152, 2016. URL http://www.nmji.in/article.asp?issn=0970-258X;year=2016;volume=29;issue=3;spage=148;epage=152;aulast=Ajami;t=6.

Jay Alammar. The illustrated bert, elmo, and co. (how nlp cracked transfer learning). URL http://jalammar.github.io/illustrated-bert/. (accessed on June 2020).

Amazon. Amazon comprehend medical, 2020a. URL https://aws.amazon.com/comprehend/medical/. Available at https://aws.amazon.com/comprehend/medical/ (accessed on November 2019).

Amazon. Supported languages, 2020b. URL https://docs.aws.amazon.com/transcribe/latest/dg/what-is-transcribe.html. Available at https://docs.aws.amazon.com/transcribe/latest/dg/what-is-transcribe.html (accessed on February 2020).

Ana Andrade. *Adoption of Electronic Health Records in the Portuguese healthcare system in the presence of privacy concerns*. PhD thesis, 2017.

auth0. Json web tokens introduction. URL https://jwt.io/introduction/.

Bartosz Skuza, Agnieszka Mroczkowska, Damian Włodarczyk. Flutter vs. react native – what to choose in 2020?, 2019. URL https://www.thedroidsonroids.com/blog/flutter-vs-react-native-what-to-choose-in-2020. Available at https://www.thedroidsonroids.com/blog/flutter-vs-react-native-what-to-choose-in-2020 (accessed on February 2020).

Jeffery L. Belden, Richelle J. Koopman, Sonal J. Patil, Nathan J. Lowrance, Gregory F. Petroski, and Jamie B. Smith. Dynamic electronic health record note prototype: Seeing more by showing less. *The Journal of the American Board of Family Medicine*, 30(6):691–700, 2017. ISSN 1557-2625. doi: 10.3122/jabfm.2017.06.170028. URL https://www.jabfm.org/content/30/6/691.

Larry Beresford. Research shows link between ehr and physician burnout. april 2016.

Brita Belli. Study: Doctors give electronic health records an 'f', 2019. URL https://medicalxpress.com/news/2019-11-doctors-electronic-health.html. Available at https://medicalxpress.com/news/2019-11-doctors-electronic-health.html (accessed on January 2020).

Rachael Brock. How to write in the medical notes. *BMJ*, 353, 2016. doi: 10.1136/sbmj.h5703. URL https://www.bmj.com/content/353/sbmj.h5703.

Jason Brownlee. What are word embeddings for text?, Aug 2019. URL https://machinelearningmastery.com/what-are-word-embeddings/.

Gabriel Catan, Rita Espanha, Rita Mendes, Orly Toren, and David Chinitz. Health information technology implementation - impacts and policy considerations: A comparison between israel and portugal. *Israel journal of health policy research*, 4:41, 08 2015. doi: 10.1186/s13584-015-0040-9.

Tiago Colicchio and James Cimino. Clinicians' reasoning as reflected in electronic clinical note-entry and reading/retrieval: a systematic review and qualitative synthesis. *Journal of the American Medical Informatics Association*, 12 2018. doi: 10.1093/jamia/ocy155.

Mohd Sanad Zaki RizviA computer science graduate. What is bert: Bert for text classification, Jun 2020a. URL https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/.

Mohd Sanad Zaki RizviA computer science graduate. What is bert: Bert for text classification, Jun 2020b. URL https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/. (accessed on June 2020).

Continuum. How soap notes paved the way for modern medical documentation, 2020. URL https://www.carecloud.com/continuum/how-soap-notes-paved-the-way-for-modern-medical-documentation/. Available at https://www.carecloud.com/continuum/how-soap-notes-paved-the-way-for-modern-medical-documentation/ (accessed on June 2020).

João Costa. *AutoSpeech: Automatic Speech Analysis of Verbal Fluency for Older Adults*. PhD thesis, 2019.

Thiago Ferreira de Toledo, Huei Diana Lee, Newton Spolaôr, Cláudio Saddy Rodrigues Coy, and Feng Chung Wu. Web system prototype based on speech recognition to construct medical reports in brazilian portuguese. *International Journal of Medical Informatics*, 121:39 – 52, 2019. ISSN 1386-5056. doi: https://doi.org/10.1016/j.ijmedinf.2018.10.010. URL http://www.sciencedirect.com/science/article/pii/S1386505618302879.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

Son Doan, Mike Conway, Tu Phuong, and Lucila Ohno-Machado. Natural language processing in biomedicine: A unified system architecture overview. *Methods in molecular biology (Clifton, N.J.)*, 1168, 01 2014. doi: 10.1007/978-1-4939-0847-9_16.

Syed Muhammad Faizan. *Applying Speech Recognition and Language Processing Methods to Transcribe and Structure Physicians' Audio Notes to a Standardized Clinical Report Format*. PhD thesis, 2020.

Naleef Fareed, Gloria J. Bazzoli, Stephen S. Farnsworth Mick, and David W. Harless. The influence of institutional pressures on hospital electronic health record presence. *Social Science Medicine*, 133:28 – 35, 2015. ISSN 0277-9536. doi: https://doi.org/10.1016/j.socscimed.2015.03.047. URL http://www.sciencedirect.com/science/article/pii/S0277953615002002.

Dieter Fensel. *Ontologies*, pages 11–18. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-662-04396-7. doi: 10.1007/978-3-662-04396-7_2. URL https://doi.org/10.1007/978-3-662-04396-7_2.

Rohith Gandhi. Naive bayes classifier, May 2018. URL https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c.

Dr. Michael J. Garbade. A simple introduction to natural language processing, Oct 2018a. URL https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32.

Dr. Michael J. Garbade. A simple introduction to natural language processing, Oct 2018b. URL https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32.

Adam Geitgey. Natural language processing is fun!, Sep 2019. URL https://medium.com/@ageitgey/natural-language-processing-is-fun-9a0bff37854e.

Google. What is a colaboratory? URL https://colab.research.google.com/. (accessed on June 2020).

Google. Streaming, 2020. Available at https://cloud.google.com/speech-to-text/docs/basics,.

GoogleCloudPlatform. Googlecloudplatform/android-docs-samples. URL https://github.com/GoogleCloudPlatform/android-docs-samples/tree/master/speech/Speech.

Shasha Han, Tait D. Shanafelt, Christine A. Sinsky, Karim M. Awad, Liselotte N. Dyrbye, Lynne C. Fiscus, Mickey Trockel, and Joel Goh. Estimating the Attributable Cost of Physician Burnout in the United States. *Annals of Internal Medicine*, 170(11):784–790, 06 2019. ISSN 0003-4819. doi: 10.7326/M18-1422. URL https://doi.org/10.7326/M18-1422.

HealthIT. What is an electronic health record (ehr)?, 2019. URL https://www.healthit.gov/faq/what-electronic-health-record-ehr. Available at https://www.healthit.gov/faq/what-electronic-health-record-ehr (accessed on June 2020).

Geoffrey Hinton, li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Phuongtrang Nguyen, Tara Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29:82–97, 11 2012. doi: 10.1109/MSP.2012.2205597.

Rani Horev. Bert explained: State of the art language model for nlp, Nov 2018. URL https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

Jeremy Howard and Sebastian Ruder. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146, 2018. URL http://arxiv.org/abs/1801.06146.

Minh Huynh, Prashant Ghimire, and Donny Truong. Hybrid app approach: Could it mark the end of native app domination? *Issues in Informing Science and Information Technology*, 14:049–065, 01 2017. doi: 10.28945/3723.

IBM. Supported languages, 2020a. URL https://cloud.ibm.com/docs/assistant?topic=assistant-language-support. Available at https://cloud.ibm.com/docs/assistant?topic=assistant-language-support (accessed on February 2020).

IBM. Supported languages, 2020b. URL https://cloud.google.com/speech-to-text/docs/languages. Available at https://cloud.google.com/speech-to-text/docs/languages (accessed on February 2020).

Emmanouil Ikonomakis, Sotiris Kotsiantis, and V. Tampakas. Text classification using machine learning techniques. *WSEAS transactions on computers*, 4:966–974, 08 2005.

Infarmed. Lista de mnsrm. URL https://www.infarmed.pt/web/infarmed/entidades/licenciamentos/locais-de-venda-mnsrm/lista_de_mnsrm. (accessed on February 2020).

Tomasz Jadczyk, Oskar Kiwic, Raj M. Khandwalla, Krzysztof Grabowski, Slawomir Rudawski, Przemyslaw Magaczewski, Hafidha Benyahia, Wojciech Wojakowski, and Timothy D. Henry. Feasibility of a voice-enabled automated platform for medical data collection: Cardiocube. *International Journal of Medical Informatics*, 129:388 – 393, 2019. ISSN 1386-5056. doi: https://doi.org/10.1016/j.ijmedinf.2019.07.001. URL http://www.sciencedirect.com/science/article/pii/S1386505619303417.

Zeqian Ju, Subrato Chakravorty, Xuehai He, Shu Chen, Xingyi Yang, and Pengtao Xie. Covid-dialog: Medical dialogue datasets about covid-19. *https://github.com/UCSD-AI4H/COVID-Dialogue*, 2020.

E. Kapetanios, D. Tatar, and C. Sacarea. *Natural Language Processing: Semantic Aspects*. CRC Press, 2013. ISBN 9781466584976. URL https://books.google.pt/books?id=Wm3SBQAAQBAJ.

Oleksii Kharkovyna. Natural language processing (nlp): Top 10 applications to know, Dec 2019. URL https://towardsdatascience.com/natural-language-processing-nlp-top-10-applications-to-know\protect\discretionary{\char\hyphenchar\font}{}{}b2c80bd428cb.

Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. Natural language processing: State of the art, current trends and challenges. 08 2017.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. 5, 11 2004.

Kundan Krishna, Sopan Khosla, Jeffrey P. Bigham, and Zachary C. Lipton. Generating soap notes from doctor-patient conversations, 2020.

Sangil Lee, Nicholas M Mohr, W Nicholas Street, and Prakash Nadkarni. Machine Learning in Relation to Emergency Medicine Clinical and Operational Scenarios: An Overview. *The western journal of emergency medicine*, 20(2):219–227, 02 2019. ISSN 0003-4819. doi: 10.5811\%2Fwestjem.2019.1.41244.

Leslie Lenert. Toward medical documentation that enhances situational awareness learning. *AMIA Annual Symposium Proceedings*, 2016:763–771, 02 2017.

Katelin M. Lisenby, Miranda R. Andrus, Cherry W. Jackson, T. Lynn Stevenson, Shirley Fan, Philippe Gaillard, and Dana G. Carroll. Ambulatory care preceptors' perceptions on soap note writing in advanced pharmacy practice experiences (appes). *Currents in Pharmacy Teaching and Learning*, 10(12):1574 – 1578, 2018. ISSN 1877-1297. doi: https://doi.org/10.1016/j.cptl.2018.09.002. URL http://www.sciencedirect.com/science/article/pii/S1877129718300261.

Alexander Mathioudakis, Ilona Rousalova, Ane Gagnat, Neil Saad, and Georgia Hardavella. How to keep good clinical records. *Breathe*, 12:369–373, 12 2016. doi: 10.1183/20734735.018016.

Microsoft. Supported languages, 2020. URL https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/language-support. Available at https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/language-support (accessed on February 2020).

Mohammad H Mobasheri, Dominic King, Maximilian Johnston, Sanjay Gautama, Sanjay Purkayastha, and Ara Darzi. The ownership and clinical use of smartphones by doctors and nurses in the uk: a multicentre survey study. *BMJ Innovations*, 1(4):174–181, 2015. ISSN 2055-8074. doi: 10.1136/bmjinnov-2015-000062. URL https://innovations.bmj.com/content/1/4/174.

MonkeyLearn. Text classification, 2020. URL https://monkeylearn.com/text-classification/. Available at https://monkeylearn.com/text-classification/ (accessed on February 2020).

João M. Monteiro and Carla Teixeira Lopes. Healthtalks - a mobile app to improve health communication and personal information management. In *Proceedings of the 2018 Conference on Human Information Interaction Retrieval*, CHIIR '18, page 329–332, New York, NY, USA,

2018. Association for Computing Machinery. ISBN 9781450349253. doi: 10.1145/3176349.3176894. URL https://doi.org/10.1145/3176349.3176894.

Janakiram MSV. Google's grpc: A lean and mean communication protocol for microservices, Jun 2019. URL https://thenewstack.io/grpc-lean-mean-communication-protocol-microservices/.

MTHelpLine. Welcome to mtsamples. URL https://www.mtsamples.com/. (accessed on January 2020).

Kim Myrick. Using clinical documentation improvement to improve patient care, Feb 2019. URL https://www.longwoods.com/content/25772//using-clinical-documentation-improvement-to-improve-patient-care.

NextGen. Understanding emr vs. ehr, 2020. URL https://www.nextgen.com/insights/emr-vs-ehr/emr-vs-ehr. Available at https://www.nextgen.com/insights/emr-vs-ehr/emr-vs-ehr (accessed on June 2020).

Nuance. Ambient clinical intelligence, 2020. URL https://www.nuance.com/healthcare/ambient-clinical-intelligence.html. Available at https://www.nuance.com/healthcare/ambient-clinical-intelligence.html (accessed on November 2019).

Office of the National Coordinator for Health Information Technology. Office-based physician electronic health record adoption, 2019. URL https://dashboard.healthit.gov/quickstats/pages/physician-ehr-adoption-trends.php. Available at https://dashboard.healthit.gov/quickstats/pages/physician-ehr-adoption-trends.php (accessed on January 2020).

Patricia Pearce, Laurie Anne Ferguson, Gwen George, and Cynthia Langford. The essential soap note in an ehr age. *The Nurse Practitioner*, 41:29–36, 02 2016. doi: 10.1097/01.NPR.0000476377.35114.d7.

Fred Pennic. Suki raises $20m to expand ai-powered, voice-enabled digital assistant for doctors, Mar 2020. URL https://hitconsultant.net/2020/03/04/suki-series-b-funding-digital-assistants/.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proc. of NAACL*, 2018.

Physiopedia. Soap notes — physiopedia,, 2019. URL https://www.physio-pedia.com/index.php?title=SOAP_Notes&oldid=211589. [Online; accessed 3-July-2020].

Sara Pinto. *Processamento de Linguagem Natural e Extração de Conhecimento*. PhD thesis, 2015.

Podder V, Lew V, Ghassemzadeh S. Soap notes, 2020. URL https://www.ncbi.nlm.nih.gov/books/NBK482263/. Available at https://www.ncbi.nlm.nih.gov/books/NBK482263/ (accessed on June 2020).

Dr Lewis Potter·Documentation. How to document a patient assessment (soap), Feb 2019. URL https://geekymedics.com/document-patient-assessment-soap/.

Practicefusion. Electronic medical notes, 2020. URL https://www.practicefusion.com/medical-notes/. Available at https://www.practicefusion.com/medical-notes/ (accessed on June 2020).

Praveen Kumar S. Analysis of native and cross-platform methods for mobile application development, 2014. URL https://www.tavant.com/sites/default/files/download-center/Analysis_of_Native_and_Cross-Platform_Methods_For_Mobile_Application_Development.pdf. Available at https://www.tavant.com/sites/default/files/download-center/Analysis_of_Native_and_Cross-Platform_Methods_For_Mobile_Application_Development.pdf (accessed on February 2020).

pythonbasics. What is flask python. URL https://pythonbasics.org/what-is-flask-python/. (accessed on May 2020).

Biao Qin, Yuni Xia, Sunil Prabhakar, and Yi-Cheng Tu. A rule-based classification algorithm for uncertain data. pages 1633–1640, 03 2009. doi: 10.1109/ICDE.2009.164.

K. Rajput, G. Chetty, and R. Davey. Phis (protected health information) identification from free text clinical records based on machine learning. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–9, Nov 2017. doi: 10.1109/SSCI.2017.8285286.

B. Raghavendhar Reddy and E. Mahender. Speech to text conversion using android platform.

Javier Rodriguez-Vera, Yenny Marin, C Borrachero, and E Pujol. Illegible handwriting in medical records. *Journal of the Royal Society of Medicine*, 95:545–6, 12 2002. doi: 10.1258/jrsm.95.11.545.

Karen R. Sando, Elizabeth Skoy, Courtney Bradley, Jeanne Frenzel, Jennifer Kirwin, and Elizabeth Urteaga. Assessment of soap note evaluation tools in colleges and schools of pharmacy. *Currents in Pharmacy Teaching and Learning*, 9(4):576 – 584, 2017. ISSN 1877-1297. doi: https://doi.org/10.1016/j.cptl.2017.03.010. URL http://www.sciencedirect.com/science/article/pii/S1877129716302180.

scikit-learn developers. Naive bayes. URL https://scikit-learn.org/stable/modules/naive_bayes.html. (accessed on June 2020).

Segen's Medical Dictionary. Health record, 2011. URL https://medical-dictionary.thefreedictionary.com/health+record. Available at https://medical-dictionary.thefreedictionary.com/health+record (accessed on June 2020).

Ji-Hyun Seo, Hyun-Hee Kong, Sun-Ju Im, Hyerin Roh, Do-Kyong Kim, Hwa-ok Bae, and Young-Rim Oh. A pilot study on the evaluation of medical student documentation: assessment of soap notes. *Korean Journal of Medical Education*, 28, 03 2016. doi: 10.3946/kjme.2016.26.

Imran Sheikh, Irina Illina, Dominique Fohr, and Georges Linarès. Learning word importance with the neural bag-of-words model. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 222–229, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-1626. URL https://www.aclweb.org/anthology/W16-1626.

Koo Ping Shung. Accuracy, precision, recall or f1?, Apr 2020. URL https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9. (accessed on June 2020).

Christine Sinsky, Lacey Colligan, Ling Li, Mirela Prgomet, Sam Reynolds, Lindsey Goeders, Johanna Westbrook, Michael Tutty, and George Blike. Allocation of Physician Time in Ambulatory Practice: A Time and Motion Study in 4 Specialties. *Annals of Internal Medicine*, 165(11):753–760, 12 2016. ISSN 0003-4819. doi: 10.7326/M16-0961. URL https://doi.org/10.7326/M16-0961.

SPAssurance. How to choose the best mobile app development framework, 2018. URL https://www.sp-assurance.com/blog/How-to-Choose-the-Best-Mobile-App-Development-Framework. Available at https://www.sp-assurance.com/blog/How-to-Choose-the-Best-Mobile-App-Development-Framework (accessed on February 2020).

Chris Spencer. punctuator. URL https://pypi.org/project/punctuator/. (accessed on February 2020).

Ewan Klein Steven Bird and Edward Loper. *Natural Language Processing with Python — Analyzing Text with the Natural Language Toolkit*. O' Reilly, 2009. ISBN 978-0-596-51649-9.

Suki. Physician burnout, 2019. URL https://resources.suki.ai/home/suki-physician-burnout-ebook-final. Available at https://resources.suki.ai/home/suki-physician-burnout-ebook-final (accessed on February 2020).

Suki. Suki ai assistant, 2020. URL https://www.suki.ai/. Available at https://www.suki.ai/ (accessed on January 2020).

Nasser H. Sweilam, A.A. Tharwat, and N.K. Abdel Moniem. Support vector machine for diagnosis cancer disease: A comparative study. *Egyptian Informatics Journal*, 11(2):81 – 92, 2010. ISSN 1110-8665. doi: https://doi.org/10.1016/j.eij.2010.10.005. URL http://www.sciencedirect.com/science/article/pii/S1110866510000241.

Sarah Syed. Clinical documentation: How to document medical information well, Feb 2020. URL https://onthewards.org/how-to-document-well/. (accessed on June 2020).

TRIBAL. Cross-platform mobile development, 2011. URL https://wss.apan.org/jko/mole/Shared%20Documents/Cross-Platform%20Mobile%20Development.pdf. Available at https://wss.apan.org/jko/mole/Shared%20Documents/Cross-Platform%20Mobile%20Development.pdf (accessed on February 2020).

Venelin Valkov. Intent recognition with bert using keras and tensorflow 2, 2020. URL https://www.kdnuggets.com/2020/02/intent-recognition-bert-keras-tensorflow.html. (accessed on June 2020).

Sumithra Velupillai, Hanna Suominen, Maria Liakata, Angus Roberts, Anoop Shah, Katherine Morley, David Osborn, Joseph Hayes, Robert Stewart, Johnny Downs, Wendy Chapman, and Rina Dutta. Using clinical natural language processing for health outcomes research: Overview and actionable suggestions for future advances. *Journal of Biomedical Informatics*, 10 2018. doi: 10.1016/j.jbi.2018.10.005.

Viera, A.F.G. and Virgil, J. Uma revisão dos algoritmos de radicalização em língua portuguesa, 2007. URL http://InformationR.net/ir/12-3/paper315.html. Available at http://InformationR.net/ir/12-3/paper315.html (accessed on February 2020).

WHO. Lista nacional de medicamentos. URL https://www.who.int/selection_medicines/country_lists/CaboVerde2009.pdf?ua=1. (accessed on February 2020).

Word2Vec. Google code archive - long-term storage for google code project hosting. URL https://code.google.com/archive/p/word2vec/. (accessed on July 2020).

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.