UNIVERSIDADE DO VALE DO RIO DOS SINOS — UNISINOS
UNIDADE ACADÊMICA GRADUAÇÃO
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

YGOR ALLAN DE FRAGA

ANAMTECH:
SPEECH-BASED AUTOMATIC STRUCTURING OF MEDICAL ANAMNESIS

São Leopoldo
2021

Ygor Allan de Fraga

ANAMTECH:
SPEECH-BASED AUTOMATIC STRUCTURING OF MEDICAL ANAMNESIS

Trabalho de Conclusão de Curso apresentado como requisito parcial para a obtenção do título de Bacharel em Sistemas de Informação pela Universidade do Vale do Rio dos Sinos - UNISINOS

Advisor:
Prof. Dr. Cristiano André Costa

Co-advisor:
Prof. Me. Frederico Falcetta

São Leopoldo
2021

ANAMTECH:
SPEECH-BASED AUTOMATIC STRUCTURING OF MEDICAL ANAMNESIS

Ygor Allan de Fraga[1]
Cristiano André da Costa[2]
Frederico Falcetta[3]

Abstract: The medical history process is critical for a correct diagnosis of the patient. Filling out medical documents is costly for the doctor and can cause some conversation details to go unnoticed, resulting in a bad patient experience or a wrong diagnosis. Helping the experience of both physician and patient is the motivation behind this work. The main objective is to create an application that automatically integrates speech recognition to turn into text the interview, identify the relevant entities for the anamnesis document, and structure a digital document. The developed model (*a.k.a. Anamtech*) integrates different services to make it possible to recognize the anamnesis properly automatically. Voice recognition was used to capture the conversation between doctor and patient. Several open libraries have already transposed the audio into text. The recognized text was included in the process of identifying essential terms for anamnesis, which healthcare professionals reviewed, and an entity recognition algorithm was used to identify such information. This algorithm was previously trained according to available existing anamnesis that passed through the process of labeling. The last Anamtech component organizes all the recognized entities in a document following a defined medical standard. A complete automatic application was created and ready to use with minor interference by the physician who uses it. As the final document is divided by entities, organized with a prefix by the anamnesis phase, it would be easy to change any information contained in it. In general, the named entity recognition (*a.k.a. NER*) model, which is the heart of this project, had a precision of 85.1%, a recall of 87.6%, and an f1-score of 86.3%. In addition, metrics for each one of the entities were captured and described. The metrics related to the patient identification had the best results, whereas the ones associated with symptoms, diseases, and treatments could be identified, but some mismatches were identified due to the difficulty to classify some entities in the pre-processing.

Keywords: Speech recognition. Named entity recognition. Anamnesis. Natural Language Processing. Medical Informatics.

1 INTRODUCTION

Artificial intelligence (*a.k.a. AI*) is the capacity of computers to simulate human intelligence and make decisions without intervention. Technologies and models of AI have been used in many different fields, and the health care system also took advantage of its benefits. AI is used in both virtual and physical branches. The virtual branch aims to use algorithms to solve problems of predictions and information extraction. On the other hand, the physical branch has

[1]Information Systems Undergraduate. Email: ygordefraga@gmail.com

[2]Advisor, Unisinos Professor, Ph. D. in Computer Science from Universidade Federal do Rio Grande do Sul (2008), M. Sc. in Computer Science from Universidade Federal do Rio Grande do Sul (1997). Email: cac@unisinos.br

[3]Advisor, Unisinos Professor, M. Sc. in Pathology Universidade Federal de Ciências da Saúde de Porto Alegre (2018). Email: fredericofalcetta@unisinos.br

a goal to use objects coordinated by AI in everyday tasks (HAMET; TREMBLAY, 2017).

One practical use of AI in health is the automation of anamnesis. The anamnesis is used to drive the physician to a hypothesis and a diagnosis. It is executed on a structured interview where symptoms, medical history, family history, social context, and other information are reported to the physician by the patient (LICHSTEIN, 1990). The data captured from the questions and the physician's perception must be documented in the medical record. Unfortunately, during the conversation, this document is filled out, leading to a lack of attention to important points that will not be included in the final notes (QUIROZ et al., 2019).

To improve the assertiveness of the consultation and save the physician from manually filling out each conversation, an automated pipeline containing natural language processing (*a.k.a. NLP*) components might be a good option since research in these fields is constantly evolving. Although the evolution is remarkable, several problems are faced in using these techniques to help automate the anamnesis, such as audio recording in a noisy place, filler words, information taken from other than speech, speakers with different contents, negative sentences, summarization, and many others (QUIROZ et al., 2019).

Besides using speech recognition to transform the audio of the anamnesis into text ready to be processed, a second problem is the organization of the information contained in the anamnesis. Physicians might not follow a pattern in the conduction of the interview depending on the patient's answers, and as a result, the captured texts may not be ideal for artificial intelligence algorithms to extract information from it. Another problem related to information extraction is explanations by the physician. Many medical terms are used to detail conditions to the patient. This information should not be included in any medical record.

Named entity recognition is a sub-field of natural language processing and aims to extract entities from any text. This technique may be used in the medical report's text to extract the vital information and categorize them in a specific group (e.g., diseases and medication names). This technique is commonly used in the biomedical field. Some technologies make it possible to extract medical information from texts (*e.g., MetaMap, cTAKES*), but the major problem is the type of language that these tools receive as input. Furthermore, these tools are originated from written text, leading to the extraction of specialist terms.

Furthermore, the storage in a medical pattern is an attention point to a pipeline that aims to extract information of medical conversation. Electronic health records (*a.k.a. EHR*) are models of organization of medical documents, and this is an ideal approach to be followed as an output since any other organization may use these documents in the future. Even if a specialization of EHR (*e.g., openEHR*) is not used in the storage, at least a data-interchange format should be used to make it easy to connect different systems.

Most AI research applied to health care focuses on predicting or recognizing which diseases and behaviors in patients. This approach helps the organizations have good performance and to give better and faster treatment to the patients. Nevertheless, studies related to the automation of bureaucratic processes are not the priority of the researchers. An example of a process is filling out a medical record while giving attention to a patient. This is a challenge for the physician since information may be lost, and the patient might feel uncomfortable with the situation. However, good management of the medical record is crucial to have cohesive data that can serve for future analysis by physicians and scientists for future research and analysis of past cases (HAMET; TREMBLAY, 2017).

Medical texts taken from anamnesis are organized as unstructured text and may or may not have a writing pattern. This type of format is not helpful for machine learning (*a.k.a. ML*) systems since algorithms need structured data to be trained. In the same way, the health professional can better understand medical records when they are structured. Therefore, research into the use of natural language processing is an increasing necessity for healthcare because of the need for valuable data for many different cases (JIANG et al., 2017).

The motivation behind this research was to make sure that the physician is concerned only with the most critical aspects of the consultation and that they do not have to waste time and attention on bureaucratic issues of filling out medical records. Along with the acceleration and improvement in the conduction of the conversation, sharing the information of the anamnesis in a digital format to other physicians and organizations can help the patient in future consultations. This research aimed to implement a model that captures the anamnesis interview using speech recognition tools and extracts the essential entities in the conversation with the support of artificial intelligence algorithms to structure a final document using a model befitting with the input. Pure JSON was chosen to facilitate the integration with any system. Along with the pipeline's construction, the proposal of a pattern in the organization of the anamnesis was included. More specific goals were recognizing the patient's information to the physician in the spoken anamnesis, extracting relevant clinical information from the anamnesis text, and structuring the extracted information according to an anamnesis pattern.

In this way, the leading scientific contribution of this work is the definition of a pattern to be followed in the anamnesis interview and the creation of a complete automatic application that recognizes the anamnesis interview, identifies the valuable information, and structures it into a digital format. The pattern helps the extraction of useful information from the anamnesis by any NER model. Furthermore, the application is prepared to work with less interference by the physician as possible, being an excellent tool to be used in any situation. Finally, as the application's output is JSON, it is possible to modify it as much as it is desired. The related work found did not show any of these two points. Only one work had some pattern in the interview, but none integrated a speech recognition module, the extraction of entities, and the structuring of information into a digital format.

This document is divided as follows: In section 2, the main principles are presented to help understand the basis of the research and the proposed model. Section 3 focuses on the description of the related papers. The proposed model is introduced in section 4, describing all pipeline stages to transform the conversation in texts, extract meaningful information, and structure the document using a medical pattern. Section 4 details the methods and development of the research, also the metrics used are present in this section. The fifth section shows the results and its discussion. Lastly, section 6 concludes the research and debates future improvements.

2 BACKGROUND

This section aims to present the theoretical basis of the research, including the definition of principles and terms such as anamnesis, natural language processing, speech recognition, named entity recognition, and electronic health records.

Anamnesis is the first presented topic in this section. Then, all the relevant aspects of the conduction of the anamnesis will be addressed. The definition of deep learning (a.k.a. DL) is discussed in the second section of the background section. Many algorithms of medical informatics use a deep learning approach. The third section introduces the ideas behind natural language processing and two critical topics for this research, speech recognition and named entity recognition. The last section introduces the concept and use of electronic health records worldwide.

2.1 Anamnesis

The diagnosis of the patient is made through the history and clinical examination by health professionals. This process is divided into two stages, the first is called anamnesis, and the second is the physical examination. Anamnesis is defined as an interview conducted by the physician during the consultation to obtain information regarding the patient's current complaint, medical history, social context, family medical history, and systems status. This is necessary to understand the patient's condition and for the proper referral of subsequent tests. In addition, a precise interview may provide focus to both exams needed and hypothesis, saving time and utilization of resources (LICHSTEIN, 1990).

Therefore, the interview is conducted in the stages of the patient's chief complaint, history of present illness (*a.k.a. HPI*), past medical story, family history, social history, and review of systems (LICHSTEIN, 1990). The patient's chief complaint gathers information about the principal problem described by the patient. HPI detail the course of the related problem, and some components like chronology, location, analogies, intensity, context, aggravating and alleviating factors, and associated manifestations are part of this. Past medical history provides a reference to the health of the patient. Family history might indicate where the problem comes from. The social history of the patient is vital to understanding habits and context. Last, questions about each system of the body are asked to assist the diagnosis.

According to Lichstein (1990), different types of content are observed in the described process. Transmitted data by the patient, where this person explains the topics related to the anamnesis process, is the raw data and will suffer some correction to explain the situation better. This edited information is known as factual content, which is the information written in the medical record by the physician. Besides the information given by the patient, observations by the physician are also used in this document, being essential to have an accurate diagnosis afterward. The interviewer may notice expressions and gestures of the patient related to the illness. The essence of the diagnosis depends on the relationship between patient and physician, so the format of the questions and empathy are essential points to gather as much of these two types of information as possible.

All the points discussed in the anamnesis are part of the medical record, which is defined as a set of documents that show a patient's consultation history. The medical record contains the anamnesis, physical examination, diagnosis, laboratory and image results, prescription, and consent terms. Thus, the anamnesis might adopt artificial intelligence in two stages: identifying the conversation between the physician and the patient and extracting relevant information in these texts. More specifically, Neural Networks, along with Deep Learning, can identify hidden patterns when a vast amount of data is given to the model. This characteristic fits well with NLP, and consequently, with the two stages previously mentioned.

2.2 Deep Learning and Neural Networks

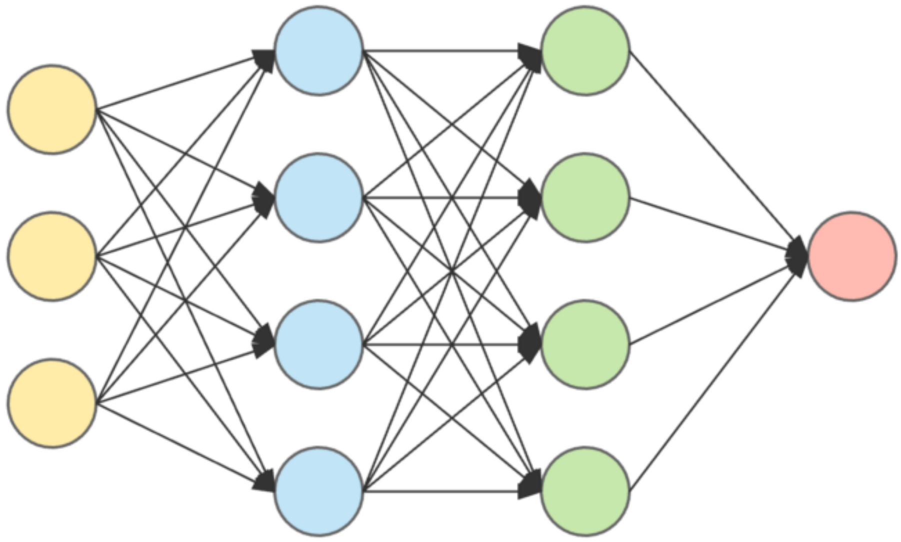
Deep learning is a machine learning field that may be used in classification, clustering, and regression. It works differently than classic machine learning algorithms. Deep Learning tries to simulate how the human brain works by analyzing the data to decide. Therefore it uses artificial neural networks to have a multi-layer structure to represent this idea.

Jiang et al. (2017) pointed that deep learning is an advanced branch of the neural network techniques since it uses many layers, which is not feasible for a classic neural network. In addition, deep learning is designed to work with a significant volume and complexity of data. The use of this conception is indicated in natural language processing, image analysis, and any other process that has many characteristics, which are not noted to be treated with classical machine learning algorithms.

As opposed to classical machine learning algorithms, deep learning does not need the manual work of feature engineering. When given a set of examples similar to the human brain, it can find the similarities of the features without manual intervention. The number of layers, functions, and variable weights must be set when any deep learning model is developed. After that, it is only necessary to provide examples for the model to be completed.

Artificial neural networks are a representation of the biological neural networks that are in the human brain. A brief explanation of the biological neural networks is required to understand better how the artificial one operates. These are formed by three structures known as cell body, dendrites, and axon. Dendrites are short forms that come from the cell body, and its responsibility is to receive signals. The axon is a supplement of the cell body, whose function is to send alerts. All the connections between neurons happen through activation produced by electrochemical pulses. The artificial neural network is a set of connected neurons, whereby

Figure 1 – Structure of a Neural Network with multiple layers



Source: (DERTAT, 2017)

these are numeric values. The relationships among all the neurons contain weights and ideally represent the same concept as an axon. The goal of training is to identify these weights, and then it is possible to execute any desired model (ELUYODE; AKOMOLAFE, 2013).

The architecture of a neural network, as shown in Figure 1, is organized into layers. The first one is called the input layer and receives a vector containing all data. There is always one neuron for each row in this layer. The layers in between are hidden layers, and they do all the math needed. The last one is the output layer and is responsible for representing the results. All of the connections between these layers have different weights defined in a matrix, which contains the weights between every neuron (WILAMOWSKI, 2009). Usually, many methods adopted in natural language processing apply deep learning algorithms to get the models to state of the art. Definitions of some methods follow.

- Convolutional Neural Networks: CNN (*a.k.a. Convolutional Neural Networks*) is primarily used in image recognition and can be used in the natural language processing field. They are defined as a composition of multiple layers that contain neurons that use filters to identify different patterns. As more profound as it goes in the convolutional layer, the patterns assigned to the neurons are more complex. The process of convolution takes a sequence of values of the input (*e.g., an image, a text*) and multiplies them by the values of each neuron, which corresponds to the pattern (DICKSON, 2020). ;

- Long Short Term Memory: This algorithm is an implementation of a Recurrent Neural Network (*a.k.a. RNN*), which according to Olah (2015) is defined as a type of neural network that has a chain of repeating components. LSTM (*a.k.a Long Short Term Memory*) is used mainly in NLP scenarios with long sequences, and where more context is needed (OLAH, 2015). Its biggest difference from RNN has a state in the cells to remember the values. All manipulation of the state is done by the gates, which are three. Forget gate is used to remove the state from the cell. The last cell output and the specific moment input determine if the value has to be forgotten. Input gate adds useful information into the state. Output gate extracts information from the current cell and transforms it into an output (ACADEMY, 2021).;

- Bidirectional Long Short Term Memory: This is an extension of LSTM that makes it possible to flow the information in both directions in the network. Bidirectional Long Short Term Memory (*a.k.a BiLSTM*) is helpful in scenarios where it is not possible to classify some information at the start of the process, but when the context is given under the course of the task, it is possible to come back in the network and determine it (VERMA, 2021).

Natural Language Processing uses several neural network techniques to identify patterns that are more complex to be recognized by classical machine learning algorithms. Analyzing the language of humans requires more advanced algorithms to treat a complex type of data that usually do not fit a linear pattern. Speech recognition and words extraction are NLP tasks that make use of neural network algorithms.

2.3 Natural Language Processing

According to Kamath, Liu e Whitaker (2019), the language analysis is organized in text and speech analysis, and either one has its categories. There are six categories for text analysis:

- Morphology: Parts of a word;
- Lexical: Identification of meaningful parts of a text;
- Syntax: Set of rules of sentences;
- Semantics: Meaning of the sentences;
- Discourse: Connection between sentences;
- Pragmatics: Characteristics of external speakers.

The speech analysis is divided into 4 categories:

- Acoustic: Representation of the sound;
- Phonetics: Production and perception of sound;

- Phonemics: Sound pattern within a language;
- Prosodics: External characteristics of a speech.

All of these characteristics, when collected, are the foundation of natural language processing. NLP is the field of computation that tries to understand the human language and interpret it (KAMATH; LIU; WHITAKER, 2019). Speech recognition and named entity recognition are essential tasks that belong to this field of study.

2.3.1 Speech Recognition

Speech recognition is a method to identify a text given an audio record. The difficulty in this topic is the different possibilities that the model might face when it works on audio recognition. For example, the speakers of the same language can speak differently depending on their culture and contexts, resulting in incorrect understanding by the model. Other characteristics that may influence the recognition accuracy are acoustic environment, volume, pronunciation, filler words, etc. Besides that, a model of speech recognition must be sturdy enough to avoid all of these issues to perform correctly (KAMATH; LIU; WHITAKER, 2019).

The acoustic model is one way of representation that seeks to translate acoustic waves into a mathematical form separated by phonemes, the smallest unit of a sound (KAMATH; LIU; WHITAKER, 2019). The author also defined Mel-frequency cepstral coefficients (*a.k.a. MFCCs*) as another type of representation that also converts waves into a vector of features.

Once a speech is translated into a text, it is possible to have a deep analysis to extract valuable information. Techniques of semantic analysis may be used to achieve this.

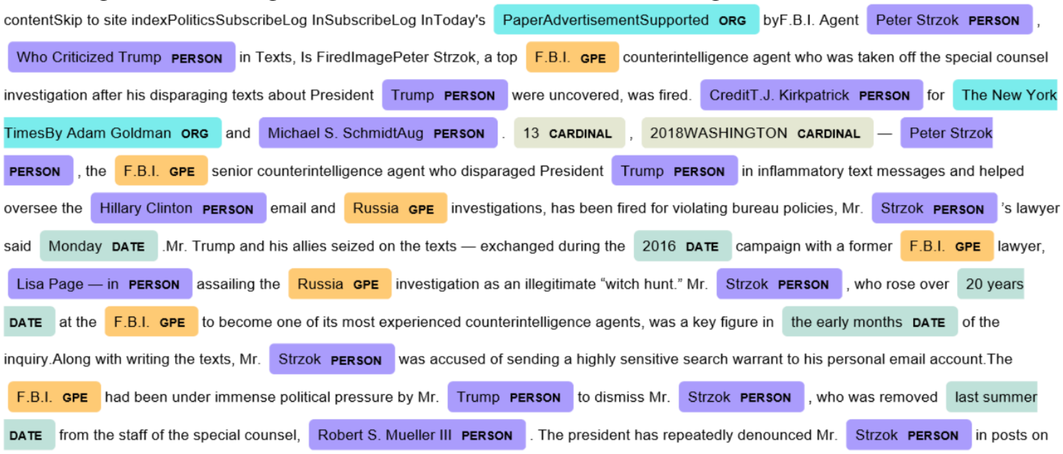
2.3.2 Word Embedding

Word embedding is one of the most critical concepts in the natural language processing area. It is defined as a vector representation of the words in some context. There are different ways to create word embedding, but the common idea is that words that appear in comparative contexts have analogous interpretations. The models usually have two tasks, the first is the text classification task, and the second one is the use of this classification to help named entity recognition (LAI et al., 2016).

2.3.3 Named Entity Recognition

Named entity recognition is a task of NLP that belongs to the semantic analysis of a text. It means that its objective is to capture meaning from the language. More precisely, NER proposes to extract entities in a given text. It is only possible because the semantic analysis first understands the meaning of the words, and then the relationship between sentences (KAMATH;

Figure 2 – Assignment of entities in a dataset using a text annotation tool



Source: (LI, 2018)

LIU; WHITAKER, 2019)

Entities are everything that might exist in the real world, such as a person, location, dates, or values. The representation of entities in a text is shown in Figure 2. Ambiguity is the most common problem found since some words found may belong to different contexts (*e.g., washington*).

Some techniques may be included in a pipeline of NER to enrich the analysis:

- Relation Extraction: Relationship between the entities;
- Event Extraction: Occurrences in a text;
- Semantic Role Labeling: Define roles to both words and sentences.

There are three different approaches when using named entity recognition (GOYAL; GUPTA; KUMAR, 2018):

- Rule-based: Only the properties of the language are used;
- Learning-based: The algorithm is trained with labeled output defined by humans;
- Feature-inferring neural network: Deep learning automatically finds the important features.

Using this method to gather important information from an anamnesis process is interesting because the return of such pipelines is regularly the name of the entities. Consequently, it is intuitive and straightforward to structure these values in documents that already expect the same entities. For example, the production of Akhtyamova et al. (2020) chooses a NER library because of its integrity with standard medical documents.

Electronic documents can be used as output from NLP pipelines, which extract information from texts. JSON files can be used as output since the entities can be considered key in this case. However, there are already created medical document standards that can be integrated with NER pipelines and generate electronic documents.

2.4 Electronic Health Records

Clinical data historically remained in silos, causing difficulty sharing information between physicians and patients (LESLIE, 2007). Moreover, different from other areas (*e.g., banking*), sharing information of hospitals is more complex. Therefore, the transmitted data need a standard, which is not straightforward in a clinical context because the structure of the processes needs to be adequate according to the proposal.

An EHR is an electronic version of a medical document usually handwritten by physicians, nurses, and health professionals. The purpose of this document is to be focused on data protection and propagation. It may contain information about the diagnosis, anamnesis, treatments, test results, and others. EHR may be considered a significant advance in health since allowed people may access it at any time, and this possibility is not feasible in a process that works primarily with physical paper (AJAMI; ARAB-CHADEGANI, 2013).

The document can be shared with different organizations and health systems, causing a gain of performance in treating either the same patient or others that may be impacted by the information contained in the EHR. Organizations also see a tremendous amount of money being saved with this approach and the improvement of the health process (AJAMI; ARAB-CHADEGANI, 2013). Using EHR is a requirement for hospitals in the United States of America (AGUIRRE et al., 2019). The organizations may also use a legacy system that replaces an EHR system. The lack of implementation of such a system will conduct the hospitals to financial penalties. Furthermore, more than improving the experience of physicians and patients, the use of EHR is a way to standardize the necessary regulations.

3 RELATED WORK

This section presents the related work of extracting medical information from texts associated with the analysis of medical texts. The base used to search related works was *Google Scholar*, and the terms of the search were words similar to *"named entity recognition in health"*, *"anamnesis speech recognition"*, *"medical conversation speech recognition"*, *"anamnesis named entity recognition"* and *"medical texts named entity recognition"*. Only papers of the last three years were considered in this exploratory research. As it is possible to notice, papers related to speech recognition were found, but it was not possible to find any work that contains the idea of recognizing a conversation and structuring the generated text.

The paper produced by Wen et al. (2021) describes a way to extract specific terms from medical documents. A named entity recognition approach was proposed based on a pre-trained modal and a dictionary of medical terms. The dictionary was built based on different sources and from the Yidu-N4K dataset, and this same dictionary was used to train the pre-trained model. The biLSTM algorithm was chosen to tune the pre-trained model. This NER approach is focused on the Chinese language. The f1-score of this method reached 95.3% in the relaxed mode of entities identification.

The work of Wulff et al. (2020) has an objective to extract meaningful information from an unstructured medical history text and represent this information in a medical pattern format, known as openEHR. This process happens through a pipeline where NLP algorithms are applied, using LingRepl to extract the information. OpenEHR has two main concepts: archetypes, the specification of a clinical topic, and templates, which are use cases based on the archetypes (e.g., anamnesis). Several archetypes were created to support the template thought. The pipeline passes through morphological analysis, POS tagging, syntactic analysis, semantics analysis, and pragmatic analysis to extract the information. After it is identified, a mapping to openEHR archetypes is executed. In addition, it has a module for handling misspelled words and denial sentences. Only precision and recall metrics were captured, with 97% and 94% recall, respectively.

The work of Akhtyamova et al. (2020) developed an NLP pipeline for the extraction of medical entities from Spanish medical texts. These entities are mainly substances, symptoms, and diseases. This study used a technique of contextualized word embedding, which is based on pre-trained models, integrated with a biLSTM algorithm to extract entities provided in the PharmacoNER. Bert and Flair were used in the word embedding stage. The results showed that the most common problem was the identification of short entities, which are common in the medical field. Recall, precision and f1-score were measured. The best performance was an f1-score of 90.84%.

The research of Cheng et al. (2019) proposed a hybrid method to identify entities and their attributes from Chinese medical texts. It uses BiLSTM in the NER task, and heuristic rules are applied to gather attributes from the entities. The texts are structured and follow 13 questions in all cases. The first task of the pipeline is the tag of the entities. Afterward, the rules are applied to identify digits, negations, time, and others. Since the attributes can be between two entities, this whole window does not have a fixed size and is treated as the potential for feature extraction. The metrics precision, recall, and f1-score were used to analyze the model. The BiLSTM-CRF method had an f1-score of 87% using the relaxed approach, which does not consider the perfect match.

Finally, the work of Śniegula, Poniszewska-Marańda e Chomątek (2019) shows a comparison study of how to identify essential terms in biomedical texts using a library called CliNER, which implements LSTM and CRF and is specified in the identification of clinical entities. A library was chosen to speed up the implementation of the model. It used the Genia Corpus to execute the NER tasks. The process contained 30 categories and 1001 records. The metrics used were accuracy, precision, recall, and F1, but the results were not shown, only the confusion matrix of the mappings.

Table 1 shows the characteristics found in the papers and the comparison between them. Different techniques were used in the pipelines of each research. Akhtyamova et al. (2020) and Wen et al. (2021) used a pre-trained model with a BiLSTM algorithm to make the NER possible. Similar to that, the work of Cheng et al. (2019) used deep learning algorithms to find the entities. The research of Wulff et al. (2020) used a series of different analyses to extract the data, such as morphological analysis, POS tagging, syntactic analysis, semantic analysis, and pragmatic analysis. A comparison of biLSTM and CRF is shown by the paper of Śniegula, Poniszewska-Marańda e Chomątek (2019).

Table 1 – Comparison between related that have a similar approach to this

	Wen (2020)	**Wulff (2020)**	**Akhtyamova (2020)**	**Cheng (2019)**	**Śniegula (2019)**
Methods	BiLSTM (NER)	NLP Tasks	Word Embeddining and BiLSTM	BiLSTM, CRF and rules	BiLSTM and CRF
Library	None	LingRep	PharmacoNER	None	CliNER
Dataset	Yidu-N4K	Private	Spanish Clinical Case Corpus	Private	Genia Corpus
Structured Text	No	No	No	Yes	No
Treatments	Spaces and Entity Boundaries	Denial and Misspelling	None	Denial	None
Language	Chinese	German	Spanish	Chinese	English
Output	No	Yes	No	No	No
Metrics	F1: 95.3%	Precision: 97% Recall: 94%	F1: 90.84%	F1: 87%	N/A

Source: Created by the author

Three of the papers used libraries to help the development of the model. LingRep was used by Wulff et al. (2020) because this library offers different components of NLP. Contrarily, Akhtyamova et al. (2020) based the development on entities and the corpus available in the PharmacoNER. This library is focused on the extraction of entities of texts written in Spanish. Lastly, CliNER was chosen by Śniegula, Poniszewska-Marańda e Chomątek (2019) for its plain focus on entity extraction of EHR.

There is a variety of languages used in the analysis of the texts. Only Śniegula, Poniszewska-Marańda e Chomątek (2019) used showed a study with texts written in English, which is the easiest language to find resources on both data and algorithms. Research of Wen et al. (2021) and Cheng et al. (2019) used Chinese records, which is considering improving its resources.

The other papers used Spanish (AKHTYAMOVA et al., 2020) and German (WULFF et al., 2020).

Only Cheng et al. (2019) adopted text with some sort of pattern. All of the data of records were structured in well-defined questions. Additionally, this paper also included treatment of negations in the answers. The work of Wulff et al. (2020) treated denials and cases of misspelling. Likely, the work of Wen et al. (2021) treated problems with spaces and entities boundaries in the pre-processing of the data. Finally, the papers of Cheng et al. (2019) and Wulff et al. (2020) used private corpus, unlike the others that used public datasets. The work of Wulff et al. (2020) is the only one that proposed an output following a medical pattern. All other papers are focused on the score instead of output.

All of the selected articles are important to understand the progress of the analyses of medical texts. At first, one may think that Wulff et al. (2020) is out of place because of its elaborations, but it presents interesting characteristics, such as the use of a word vectorization technique and structured representation of medical information. On the other hand, other papers focus on using NER to extract medical information from structured or unstructured medical text. The use of libraries (*e.g., Spacy*) is quite common, as well as some kind of manual treatment of the extracted entities.

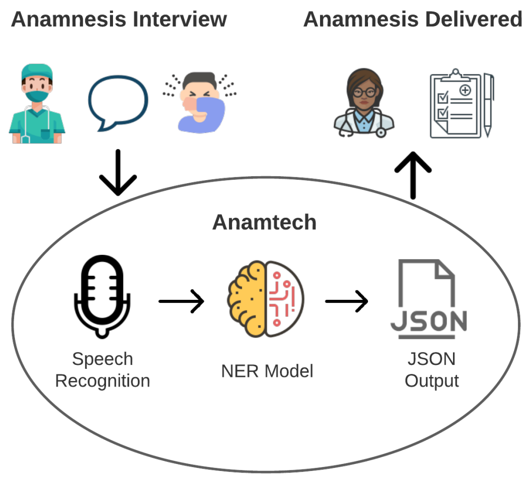
The metrics of the papers were all available, except the paper of Śniegula, Poniszewska-Marańda e Chomątek (2019), which has a section explaining the metrics gathered, but none of them appeared in the text. The rest of the researches showed the f1-score, like Wen et al. (2021), Akhtyamova et al. (2020) and Cheng et al. (2019), and the work of Wulff et al. (2020) only presented the precision and recall, which are the base of f1-score. The best performance among the works was Wulff et al. (2020), which has a precision of 97% and a recall of 94%, and the last work in terms of performance was Cheng et al. (2019) with an f1-score of 87%.

The gaps found were in the integration between these features and speech recognition. No article found uses voice recognition for anamnesis and unites the identification of entities with information structuring using either EHR or any digital output. Implementing such a model would be of great value to speed the process involving the doctor and the patient, even with the possible problems that may occur in the misinterpretation of information.

4 ANAMTECH AND ITS METHODS

This section describes the proposal of this research, the application Anamtech. Here are described the application process, the dataset used to train and test, details of pre-processing, and development of all modules.

Figure 3 – Complete flow of use of the Anamtech application



Source: Created by the author

4.1 Process Overview

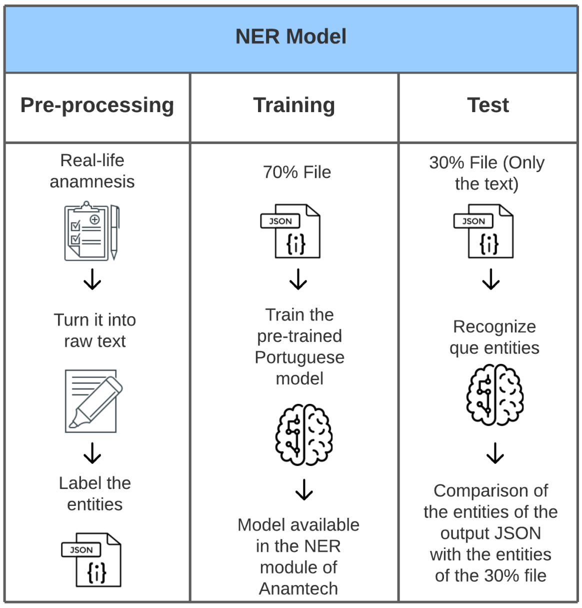
Anamtech aims to use speech recognition to identify the conversation between the doctor and the patient and consequently use an NLP library to extract the entities belonging to the text. These entities should be structured and made available to the physician digitally. A complete application was developed with different modules: the NER model, speech recognition, and output structuring. All these modules are well integrated, making it easy to use Anamtech in the entire process of anamnesis.

The focus was the quality of recognizing the valuable information in the interview. One crucial part of the development of any AI model is the pre-processing work. Processing clinical data is not simple. Furthermore, NER algorithms need labeled data in the training process, making the manual work time-consuming and vital to define the target entities. A pattern needs to be followed in the conduction of the anamnesis. It helps the NER model to identify the entities and includes them in the correct label. The focus of the application is the extraction of detailed information of the anamnesis stages, and the creation of the entities was based on this.

The use of Anamtech is defined by Figure 3. The use process starts with the recording of the anamnesis interview by the module of speech recognition included in the Anamtech. The text is then passed to a trained NER model, which identifies essential terms for the anamnesis. After the recognition by the model, an output module transforms the entities and creates a JSON file containing all the terms found. Whenever a repeated term is found, the key turns into a list of terms. The physician can edit the digital document created if needed and optionally fill a physical document with the information present in the interview.

More details of the process of the NER model are shown in Figure 4. Manual work was done to create simulated interviews from the real-life anamnesis, which were already filled in

Figure 4 – Step-by-step of the development of the NER model included into Anamtech



Source: Created by the author

a document, and to label these created texts with the correct anamnesis entity in the precise position. Thus, a total of 70% of the simulated interviews will have a document with its entities, which will serve the pre-trained Portuguese model to specialize it based on the needs of Anamtech. The 30% of the simulated interviews was used in the model test, and this test can be based only on the document with the comments of all entities. The separation of files was purely based on the order of availability. There is no visible difference between the anamnesis according to its position.

4.2 Dataset

Permission was requested to use anamnesis performed by medical students at Unisinos. These documents were created based on interviews with patients in the stage of their course where this learning takes place. All the documents have been completely anonymized, and much of the information has been edited. Personal information of the patients was changed by the experts to make their recognition impossible. Furthermore, an adaptation of the information to follow medical terms was performed. Sixty records were used in this work, and all the records were used only in the training and testing of the NER model. The accessible set of anamnesis are not conversations, but they are documents divided by the stages of the anamnesis interview (LICHSTEIN, 1990). The data format does not follow the pattern needed by the NER model since it is already organized in a similar format proposed by this study. The model needs a conversation, and the documents available were created from this conversation.

The data is available in the format of a *.docx*, and its format is shown by Figure 5. Two pieces of information were hidden, the name and age of the patient. This anamnesis document is divided into the stages of the anamnesis, and all entities mapped by the specialists are comments in the same text. The document starts with the identification stage, which is the part that has basic information about the patient, such as the name, age, profession, religion, and marital status.

It is followed by the description and time of the complaint that guided the patient to the medical visit. Standard information declared in the complaint description are pain, nausea, fever, and many other possible problems. Also, it includes the time that the problem started. After that, the next step is the HPI, which is the detailing of the illness. The course of the problem, symptoms, secondary symptoms, treatments, results, alleviation, and organs is described in this document stage.

Then the medical history is specified, where all illnesses and treatments are indicated, and the use of drugs and possible allergies. After that, the illness related to the family is outlined. This stage contains the history of illness and the age of death in the family. Next, the social aspects of the patient, such as house partner, house type, diet, and sleep, are related in the social history stage. The last stage is the review of systems, where is described the different organs and systems of the patient.

4.3 Pre-processing

Some changes had to be applied in the available anamnesis to make it functional for the NER model. It starts with creating conversations since every anamnesis follows a format that does not follow the input of the speech recognition module. After creating conversations that simulate the ready anamnesis, it is mandatory to start a process of labeling, which may be done using technologies like Doccano.

4.3.1 Interview Simulation

Every phrase and word highlighted in Figure 5 represents an entity that is mapped in a subsequent process. Simulated conversations were created from this document to be used by the NER model. This simulation contains possible words exchanged between the physician and the patient, and all entities are added to this. Other data such as filler words and greetings are also added to have the most real conversation possible. Not all information contained in all stages was applied to the simulation since experts who reviewed these documents did not mark them as entities. This simulation is necessary due to the type of input demanded by the NER model, which in the normal process of use will receive the raw text of the speech recognition, in other words, a conversation in a text format.

Figure 6 shows the simulation of the same document spoken above. The creation of the interview took into account every entity marked by the experts. The only part that had not have

Figure 5 – Example of an anamnesis document used in the development, all marked entities are colored

IDENTIFICATION: NAME, AGE anos, feminino, branca, viúva, natural de Itaqui, procedente de Esteio, cristã e aposentada.

COMPLAINT: "Desmaio"

HPI: Síncope no dia 09/05/2021 associado à vertigem. Apresentou também náuseas e sensação de calafrio. No final do ano passado apresentou quadro de dor lombar súbita, em pontada, com intensidade 9,5/10 associada a sensação febril e náuseas. Foi tratada com medicamento, mas não sabe o nome. Já em fevereiro deste ano, na UBS, recebeu o diagnóstico de cálculo renal e foi tratada com ibuprofeno e diclofenaco de 8/8h. Em março, desenvolveu úlcera e gastrite associada à febre, náuseas, sendo tratada com medicamentos naturais, mas não soube o nome dos medicamentos.

MEDICAL HISTORY: Câncer de útero diagnosticado em 2003 e tratado através de histerectomia em 2005. Câncer de mama esquerda diagnosticado em 2015 e tratada com quimioterapia e mastectomia no mesmo ano, fez uso de hormonioterapia de 2015 a 2017, não soube indicar nome e dosagem do medicamento, parou o tratamento por ter se mudado. DM tipo 2 e hipertensão diagnosticados na internação do dia 09/05/21, sem uso de medicamentos contínuos. Sem alergias e sem história de uso de álcool, tabaco ou drogas ilícitas. Cesária de urgência em 2000.

FAMILY HISTORY: Pai falecido aos 76 anos por IAM e mãe falecida aos 67 anos por câncer metastático de sítio primário desconhecido. Tem 12 irmãos 9 hígidos e 3 falecidos, por AVC, câncer de mama e AIDS. Tem 4 filhos hígidos.

SOCIAL HISTORY: Ensino fundamental completo. Mantêm alimentação balanceada, sem excesso de sal e doces e com arroz, feijão, carne branca e saladas todos os dias e mantém rotina de exercícios físicos com caminhadas diárias de 1 hora. Não consegue dormir todas as noites por insônia. Mora sozinha, tem bom relacionamento com vizinhos e os filhos e tem como hábito de lazer assistir televisão e conversar com os vizinhos.

REVIEW OF SYSTEMS:

GENERAL: Perdeu 20 kg desde dezembro,

EYES: uso de óculos, última consulta ao oftalmologista em dezembro de 2020.

EARS: dor e zumbido no ouvido esquerdo em março de 2021. Sem secreções no canal auditivo.

Source: Created by the author

all entities was the review of the system. It is crucial to notice that characters like "?", ",", and "." were avoided due to an inability of the Speech Recognizer library to represent these punctuations. Another characteristic is the lower case format of every word, which was adopted to define a pattern of both input and output.

An adequation of medical terms was demanded when transforming the complete anamnesis in interviews. The words used in the document are uncommon in any interview since they are used more often by health professionals. For instance, words like "dispnéia" and "pirose", which are equivalent to shortness of breath and heartburn, are not used by patients in a regular medical appointment, then they were changed by "falta de ar" and "azia". Many other terms were modified to have the most realistic simulation possible.

Figure 6 – Simulated anamnesis interview based on the anamnesis document

bom dia **NAME** quantos anos você tem tenho **AGE** anos é casada sou viúva ok trabalha aposentada e a religião cristã o que te trouxe aqui tive um desmaio e como iniciou isso tudo tive esse desmaio junto com vertigem também senti muitas náuseas e sensação de calafrio teve outro sintoma relacionado dor nas costas com febre ano passado esse ano também tive uma úlcera e gastrite com febre e náuseas e seu histórico médico quais problemas já teve tive câncer de útero em 2003 e tratado com histerectomia em 2005 outro câncer de mama em 2015 e tratada com quimioterapia no mesmo ano tenho hipertensão alguma alergia nenhuma usa álcool ou fuma nunca e sua família como estão meu pai morreu aos 76 anos com ataque cardíaco e a minha mãe aos 67 anos com câncer metastático 3 irmãos morreram um com avc outro com câncer de mama e outro de aids e a dieta boa sem muito sal e doces com arroz feijão carne branca e saladas todos os dias dorme bem não consigo dormir todas as noites por insônia mora sozinha sim vou revisar algumas coisas perdeu peso sim 20 quilos desde dezembro como estão os olhos eu uso óculos e a última consulta foi em 2020 entendi e o ouvido eu tenho um pouco de dor e zumbido no ouvido esquerdo desde 2021

Source: Created by the author

Figure 7 – Anamnesis labeling using Doccano

Source: Created by the author

In terms of data, there are no differences between the anamnesis in Figure 5 and the Figure 6. There is only adequacy of terms and format in the simulated interviews to turn the process of entity extraction more realistic. The approach used in the physician's and the patient's speeches were a bit different. The physician's one followed a more formal dialogue, whereas the patient's one kept an informal speak most of the time.

4.3.2 Labeling

All used entities were created in the open-source annotation tool Doccano. This tool was used to label all existing entities in the simulated interviews, and these come from the anamnesis document. The simulated interviews kept most of the entities pointed by the specialists in the anamnesis document. Doccano's input is a document containing all simulated interviews created in the last step separated by a line feed (*a.k.a. /n*). Once the process of labeling all the entities is done, as shown in Figure 7, a JSON file containing the interview text with the position and name of all entities in the text is created. The output of this annotation tool can be used by the NER model training with just type changes. Figure 8 shows an example of a JSON with only 2 entities mapped.

Every labeled entity in the simulated interview has as a prefix the name of the stage that the entity belongs to. In total, 25 entities were created for this model, and they will serve as input for the NER model and output for the JSON created in the final module of Anamtech. List of entities with a short description about their meaning:

- **Identification Name:** Full name.

- **Identification Age:** Age of the patient.
- **Identification Profession:** Current or old job position.
- **Identification Religion:** Religion if exists.
- **Identification Marital:** Current marital status.
- **Complaint Description:** Information about the complaint that took the patient to the hospital.
- **Complaint Time:** When the problem appeared.
- **HPI Course:** Complete problem appearance course.
- **HPI Symptoms:** All symptoms noticed by the patient.
- **HPI Other Symptoms:** Symptoms not related to the main complaint.
- **HPI Alleviation:** Methods used to alleviate the complaint.
- **HPI Organs::** Organs affected.
- **HPI Treatment:** Medicines and procedures used to treat the problem.
- **HPI Results::** Results of the treatment.
- **Medical History Illness:** Historic of illness.
- **Medical History Treatment:** Treatments used in past illnesses.
- **Medical History Allergy:** Allergies of the patient.
- **Medical History Habit:** Use of drugs, tobacco, and alcohol.
- **Family History Illness:** Problems in the family.
- **Family History Death Age:** Age that relatives died.
- **Social History House Partner:** People with whom the patient lives.
- **Social History House Type:** Information about habitation.
- **Social History Sleep:** Duration and quality of sleep.
- **Social History Diet:** Type of food ingested.
- **Review of Systems Status:** Revision of different parts of the body.

Figure 8 – Example of Doccano JSON output


```
{
  "data": "oi boa tarde sou a camila qual é o seu nome olá doutora me chamo
    NAME qual é a sua idade AGE é casado sou viúvo religião católico você
    trabalha sou aposentado porém eu trabalhava na lavoura e em uma firma e
    qual é a sua queixa tive crise de pressão parecia um avc quando isso
    aconteceu 2 dias atrás como tudo se iniciou senti uma perda de força de
    sensibilidade e de movimentos nos braços tive dificuldade para falar
    rosto parecia paralisado não conseguia ver nada um pouco de dor no peito
    e inicio de repente sim foi espontâneo e durou cerca de 40 minutos alguma
    outra coisa aconteceu mal-estar tontura e queda de pressão como estava a
    pressão 100 por 40 perdeu a consciência ou memória não sentiu uma melhora
    a melhora foi aparecendo do nada e seu histórico quais doenças já teve
    tive sarampo catapora e varicela na infância hoje sou hipertenso estou em
    tratamento com diurético 25mg alguma cirurgia fiz uma cirurgia no tendão
    de aquiles desvio de septo há 30 anos e hérnia há 5 anos também uma
    cirurgia torácica após um acidente em 1977 com fraturas nas costelas
    bacia e esterno comprometimento da base pulmonar esquerda também e faz ou
    fez outro tratamento alenia 12 micrograma remédio para próstata remetido
    para colesterol tenho um marca passo alguma alergia sulfa usa álcool
    cigarro ou drogas não e o histórico familiar pai morreu aos 79 anos mãe
    morreu aos 80 anos ela tinha problema no coração meu irmão morreu de
    câncer de pulmão aos 41 anos meu irmão tem 75 anos é cardiopata e
    hipertenso minha esposa morreu com problemas renais aos 74 anos um filho
    meu morreu em um acidente aos 41 anos e sobre a questão social mora com
    quem estou morando com a minha filha e a alimentação bem variado gosto
    muito de frutas e o sono 10 horas por noite vamos fazer uma revisão como
    está o peso sem alteração problemas de visão hipermetropia e miopia algo
    no pescoço nada atualmente algo no coração sem dor ou palpitação câimbra
    não e a respiração tosse quando me esforço apareceu após cirurgia
    tremores não alteração na memória sim tenho um branco quando escrevo",
  "label": [
    [
      65,
      69,
      "Identification Name"
    ],
    [
      98,
      101,
      "Identification Age"
    ]
  ]
}
```


Source: Created by the author

4.4 Development

This section describes the technologies used in developing the whole application and the implementation details of all modules that compose Anmantech, which are the NER model, speech recognition, and output structuring. Some technologies were used to develop the application based on the proposed model:

- Doccano: This library speeds the time to label entities in a text manually. All the entities have to be registered previously. Then the uploaded text can be annotated. The visualization of the text after the labeling is an advantage o this tool.

- Spacy: A library that provides NLP pipelines to build AI products. It has support in more than 64 languages, including Portuguese. Spacy also makes it possible to build customized models (*e.g., NER pipeline based on created entities*). Version 3 of Spacy was used.

- SpeechRecognition: This tool performs speech recognition with support for different APIs, being possible to use online and offline execution. The version used was 3.8.1, and Google Speech Recognition was chosen.

- Python: All the libraries described above are available in the language Python, and this was the programming language used in the development of the application. Besides using the APIs of speech recognition and NER, the identification of entities and structuring of the output will also be made using Python. The version chosen was 3.8, which integrates well with any needed API.

4.4.1 NER Model

The development of the training of the NER model is divided into the transformation of the annotation JSON in an object that contains the serialized data and the script to execute the training itself. Listing 1 shows the loading of the annotated data (Figure 8) and the transformation in a format understandable by Spacy. A blank Portuguese pipeline must be defined, and all entries processed are assigned to a Doc object containing the data and the position of every entity. Once the Doc is created, it is added to a DocBin object, an iterator of docs with a function to serialize the data in a Spacy format.

Listing 1 – Serialization of annotated data that was used in the training of the NER model

```
from json import loads
import spacy
from spacy.tokens import DocBin
from tqdm import tqdm

pre_processing_json = [loads(line) for line in open(
    "model/data/pre_processing_train.json", "r")]

nlp = spacy.blank("pt")
db = DocBin()

for line in tqdm(pre_processing_json):
    doc = nlp.make_doc(line["data"].lower())
    doc.ents = [doc.char_span(
        ent[0], ent[1], label=ent[2],
        alignment_mode="contract")
        for ent in line["label"]]
    db.add(doc)

db.to_disk("model/data/train_data.spacy")
```

After the serialization, a config file must be created to configure the model in the best way possible. This file contains information concerning the components used in the pipeline, base dataset to use, number of hidden layers, number of batches, and others that are default. The values set were chosen according to the recommendation of the Spacy community.

- **Components**: The components chosen were tok2vec and ner because both are mandatory to have a pipeline to identify entities.

- **Base Dataset**: It used 'pt_core_news_lg' dataset due to its size. It is the biggest Portuguese dataset.

- **Number of Hidden Layers**: 64 layers were used. This number of layers is the default value set by spacy's NER model.

- **Number of Batches**: The model was trained in 1000 batches. This is the recommendation for pipelines with just a few examples.

With the base config file created, it is just necessary to fill the config file with the rest of the default configurations needed by Spacy and execute the specific command to train the model passing the output path of the model and the path of the serialized data (Listing 2). The result is a serialized model that the Spacy library can load. This training process has its own metrics, but they were not considered in this study for the complexity of calculating metrics related to NER models.

Listing 2 – Script to execute the training of the NER model

```
source .venv/bin/activate
python3 -m model

python3 -m spacy init \
    fill-config model/config/base_config.cfg \
    model/config/config.cfg

python3 -m spacy train model/config/config.cfg \
    --output model/output \
    --paths.train model/data/train_data.spacy \
    --paths.dev model/data/train_data.spacy
```

In general, before applying any other component, Spacy applies the component of tokenization, which aims to create a Doc object of all words contained in the text. With this component applied, all other components may be executed (SPACY, 2021). The NER model that belongs to Spacy uses intelligent feature engineering, which extracts characteristics from the data. Before these features are added to the extractor of entities, every word receives a unique representation according to the context in which it is inserted (TERRY-JACK, 2019). The deep learning algorithm used by Spacy is CNN (*a.k.a. Convolutional Neural Networks*) (CZERWINSKA, 2019).

4.4.2 Speech Recognition

The first module that can be used by the application Anamtech is Speech Recognition. It was developed to make it possible to record the conversation between the physician and the patient and automatically use subsequent modules to extract correct entities and structure the anamnesis in a digital format. For example, Listing 3 shows the code representing the class SpeechRecognition, which was designed to start the recognition of both live conversations and audio files. However, only the live recognition was developed since it is the only necessary function to its version of Anamtech.

Listing 3 – Portuguese speech recognition module used in interview recording

```python
import speech_recognition as sr
from json import loads

recognition_options = {"MIC": "MIC"}


class SpeechRecognition():

    def __init__(self, type):
        self.type = type

    def recognize(self):
        if (self.type == recognition_options["MIC"]):
            return self._recognize_mic()
        else:
            return ""

    def _recognize_mic(self):
        rec = sr.Recognizer()

        with sr.Microphone() as mic:
            rec.adjust_for_ambient_noise(mic)
            print("Start␣recognizing␣[MIC]")
            audio = rec.listen(mic)
            try:
                return rec \
                    .recognize_google(audio, language='pt-BR')
            except sr.UnknownValueError:
```

```
                    return ""
```

Once the live recording is selected, an object of the class Recognizer is instantiated. This class contains functions representing the functionalities of the SpeechRecognizer library. The possibility to use a physical microphone is done through the instantiation of the class Microphone. With both objects, it is possible to configure the recognition through the microphone, changing some parameters, such as adjusting the ambient noise function, that calibrates the ambient noise. The function *listen* belongs to the recognizer object and takes as argument an audio source, which in this case is the microphone object. With the recording ready, it is only necessary to choose the engine and the audio language. The Google engine was chosen due to its heavy use of the Python community, and the Portuguese language was chosen. The return of this module is a string containing the recognized conversation.

Listing 4 – Entity recognition module uses the trained NER model to identify the entities

```
import spacy


class NamedEntityRecognizer():

    def __init__(self):
        self._nlp = spacy.load('model/output/model-last')

    def recognize(self, text):
        print("Identifying entities")
        return [
            {
                "text": entity.text,
                "entity": entity.label_
            }
            for entity in self._nlp(text).ents
        ]
```

4.4.3 Entities Recognition

The module that does the recognition of the entities has only one function called *recognize* that receives as input the string representing the anamnesis interview. The trained model is loaded and assigned to an internal attribute when an object of this class is instantiated. The recognize function then uses the loaded object to extract the entities. An iteration is done through the conscious entities, and the entity and its data are returned in an array of objects.

The load method of the Spacy library receives the path of the trained model. Listing 4 shows the implemented code.

Listing 5 – Logic used to structure the output in a JSON format

```
from json import loads, dump
from time import localtime, strftime

registered_entities = [
    "Identification␣Name",
    "Identification␣Age",
    "Identification␣Profession",
    "Identification␣Religion",
    "Identification␣Marital",
    "Complaint␣Description",
    "Complaint␣Time",
    "HPI␣Course",
    "HPI␣Symptoms",
    "HPI␣Other␣Symptoms",
    "HPI␣Alleviation",
    "HPI␣Organs",
    "HPI␣Treatment",
    "HPI␣Results",
    "Medical␣History␣Illness",
    "Medical␣History␣Treatment",
    "Medical␣History␣Allergy",
    "Medical␣History␣Habit",
    "Family␣History␣Illness",
    "Family␣History␣Death␣Age",
    "Social␣History␣House␣Partner",
    "Social␣History␣House␣Type",
    "Social␣History␣Sleep",
    "Social␣History␣Diet",
    "Review␣of␣Systems␣Status"
]


class AnamnesisStructure():

    def create_json(self, recognized_entities):
```

```
json = {}
for recognized_entity in recognized_entities:
    entity = recognized_entity["entity"]
    if entity in registered_entities:
        if entity not in json.keys():
            json[entity] = \
                [recognized_entity["text"]]
        else:
            if recognized_entity["text"] \
                    not in json[entity]:
                json[entity]\
                    .append(recognized_entity["text"])

for key in json.keys():
    if len(json[key]) == 1:
        json[key] = json[key][0]

final_json = {}
for ent in registered_entities:
    if ent in json.keys():
        final_json[ent] = json[ent]

date_time = strftime("%Y-%m-%d_%H:%M:%S", localtime())
path = f"output/anamnesis_{date_time}.json"

with open(path, 'w', encoding='utf-8') as f:
    dump(final_json, f, ensure_ascii=False, indent=4)

print(f"Output: {path}")
```

4.4.4 Output Structuring

The class used to structure the output of the NER model is shown by Listing 5. Only one function was created to prepare the document, which is the *create_json* function. At first, an iteration is applied in the array of recognized entities, and all entities are assigned to a JSON object where the key is the entity's name and the value is either the single value of the entity or an array of values. It may happen in cases where the model identified more than one crucial piece of information in the anamnesis. Once the JSON is created with its values and ordering is done according to a constant of entity names, where it follows the expected order of information in the anamnesis interview.

The last part is the creation of the JSON file, in which the name of the file has a suffix containing the timestamp of creation based on the BRT timezone. The file is available in an internal folder, which the doctor may access and change according to the necessity due to errors or misunderstandings by the NER model. An example of the created JSON with the complete anamnesis structure may be seen in Listing 6.

Listing 6 – JSON created in the output module following the sequence of information expected: Identification, complaint, HPI, medical history, family history, social history and review of systems


```
{
    "Identification␣Name": "Lourdes␣Marta␣Rabello",
    "Identification␣Age": "74",
    "Identification␣Profession": "aposentada",
    "Identification␣Marital": "viuva",
    "Complaint␣Description": "desmaio",
    "Complaint␣Time": "4␣horas␣atras",
    "HPI␣Symptoms": "dor",
    "HPI␣Organs": "parte␣das␣costas",
    "Medical␣History␣Illness": [
        "catapora␣caxumba␣sarampo␣e␣coqueluche",
        "infeccao␣urinaria",
        "diabetica",
        "hipertensao␣nao"
    ],
    "Medical␣History␣Treatment": [
        "hospital",
        "insulina",
        "aas",
        "clinica"
    ],
    "Medical␣History␣Allergy": "penicilina",
    "Family␣History␣Illness": [
        "avc",
        "cancer␣de␣garganta",
        "diabetes"
    ],
    "Family␣History␣Death␣Age": [
        "71",
```

```
        "70"
    ],
    "Social␣History␣Sleep": "9␣horas",
    "Review␣of␣Systems␣Status": "cansaco␣sim"
}
```

4.4.5 Modules Connection

All modules were connected in one single execution calling directly the file *run.py* which belongs to the Python module *application*. The function *main* may have only one argument, which is the text to have its entities extracted. If no argument is passed, then the speech recognition module is called, and the extraction of entities is done after recording the interview. It was thought that way to make it easier to test the model with the simulated interviews created. Listing 7 shows the connection between the modules speech recognition (optional), NER model, and output structuring.

Listing 7 – Main function that connects all modules

```python
import argparse
from json import loads
from helpers *

parser = argparse.ArgumentParser()
parser.add_argument("-t", "--text",
                    help="Anamnesis␣to␣be␣structured")
text = parser.parse_args().text

if text is None:
    recognition_options = loads(
        open("constants/recognition_options.json", "r").read())
    text = SpeechRecognition(recognition_options["MIC"])\
        .recognize()

entities = NamedEntityRecognizer().recognize(text)
AnamnesisStructure().create_json(entities)
```

4.5 Metrics

The focus of the work evaluation was on the NER model and all the entities separately since it is the core of the Anamtech application, and another module that could have been analyzed was the speech recognition module, but this is a more auxiliary module that uses a third-party library used worldwide. As it was possible to notice, some metrics such as precision, recall, and f1-score are available in the model's training process. However, these metrics were not taken into account because of the complexity of the determination of the entities. For example, the entities can be texts many times, and one or two words should not interfere with its evaluation. Then to avoid such a problem, all of these three metrics were analyzed manually using 30% of the simulated interviews and the comments by health specialists to determine the possible outcomes: true positives, false positives, true negatives, and false negatives.

The metrics precision, recall, and f1-score were analyzed per entity individually and per the whole model. In this study, precision is the percentage of correct entities found by the model. The entities were considered correct when the value was at least partially correct since the use case of this research has many entities which are answers. A large number of entities were not considered true positives if one word was missing. The metric recall is the percentage of the correct entities present in the dataset identified by the model. F1-score is the mean between the metrics recall and precision (SHUNG, 2018). The formulas to calculate the metrics are:

- **Precision**: True Positives / (True Positives + False Positives).
- **Recall**: True Positives / (True Positives + False Negatives).
- **F1-Score**: 2 * (Precision * Recall) / (Precision + Recall).

Before calculating these metrics, it was necessary to define the outcomes. The definition can be true positive, true negative, false positive, and false negative. The positive and negative are related to the class classified by the model (*e.g., is good, is not good*), whereas true and false give the information of the capacity by the model to classify correctly any record or information (MISHRA, 2021). In this use case, classes are not used, so the concept had to be adapted. The definition of these terms in this evaluation are below:

- **True Positives**: Information that was partially assigned to the correct entity. It has the same meaning when analyzing every entity individually and the NER model as a whole.
- **False Positives**: Information identified as entities that the specialists did not mark. Some entities may be false positives in one entity but a false negative in another. The NER model will have only one more false positive and not another false negative in these cases.
- **False Negatives**: Information marked as entities by the specialists that the model did not recognize.

True negatives do not apply here. The understanding of a correctly labeled entity was based on the entities' comments by the specialists in the document of the anamensis. The simulated interview does not compose all entities marked in the document, so just the entities contained in the used text were considered.

5 RESULTS AND DISCUSSION

Table 3 shows the number of true positives, false positives, and false negatives found in each of the established entities and the NER model. As explained in a past section, the definition of false negatives was different for the individual entities and the general model. The confusion matrix is available on Table 2. Only the confusion matrix of the NER model was created, and as there are no true negatives in this use case, this quadrant has the value 0. Table 4 shows the metrics precision, recall, and f1-score, which were based on the results of Table 3. To have a better visualization about the differences of the three metrics among all the entities, the Figure 11, Figure 9 and Figure 10 represent the growing value of the f1-score, precision, and recall of all entities analyzed in a graphic format.

Table 2 – Confusion Matrix of the NER model

	Predicted Positive	**Predicted Negative**
Actual Positive	463	65
Actual Negative	81	0

Source: Created by the author

All entities of the identification section had a good performance. As the anamnesis followed a specific pattern, and all entities grouped in the identification section (name, age, profession, religion, and marital) usually have one value, it had an expected classification. The lowest f1-score was 84.8% for the patient's name, and the biggest was 98.8% for the patient's age. The name had two false positives and three false negatives. The false positives found were always the doctor's name, which appeared in some conversations, and false negatives had either an unusual name or a filling conversation before the patients provided their name, showing a generalization problem in some cases. The age entity had no false negatives and only one false positive. In all conversations used for testing and training, no other number was found in the interview's introduction.

Table 3 – True positives, false positives and false negatives found by entity and by the model

	True Positives	**False Positives**	**False Negatives**
Identification Name	14	2	3
Identification Age	18	1	0
Identification Profession	20	4	1
Identification Religion	9	0	2
Identification Marital	17	1	1
Complaint Description	16	0	1
Complaint Time	14	3	4
HPI Course	7	3	6
HPI Symptoms	15	4	4
HPI Other Symptoms	24	5	17
HPI Alleviation	3	4	4
HPI Organs	3	2	9
HPI Treatment	5	1	6
HPI Results	3	1	3
Medical History Illness	48	11	10
Medical History Treatment	55	12	6
Medical History Allergy	10	1	4
Medical History Habit	20	0	9
Family History Illness	35	2	8
Family History Death Age	21	1	7
Social History House Partner	13	3	0
Social History House Type	13	0	3
Social History Sleep	14	4	2
Social History Diet	21	0	1
Review of Systems Status	45	16	11
NER Model	463	81	65

Source: Created by the author

Complaint and HPI sections are correlated since both sections describe the problem that brought the patient to the hospital. The entity related to the description of the complaint had no false positives and only one false negative. Both recall and precision had good performances.

On the other side, the information about the time of the complaint had many true positives, but in some cases had false positives and false negatives. The three false negatives found may be explained because of the different descriptions of the problem's time. Since this research had 48 conversations used for testing, the number of false positives, in this case, should decrease because of the different descriptions of time. The four were visibly confused with the entity *HPI Course*, which is also related to time. Therefore, the f1-score for both entities of the complaint section had a regular performance.

HPI is the section with more entities. All of them are related to the course of the complaint, and they range from time to treatment of the problem. *HPI Course* was one entity that was hard to define by the specialists. Sometimes it was defined like specific times of the disease, and other times as a complete sentence showing all patient steps. As expected, this entity had a precision (70%) more prominent than the recall (53.8%) due to the definition problem in some cases. The entities associated with the symptoms, *HPI Symptoms* and *HPI Other Symptoms*, had both a good precision, 78.9%, and 82.7%, respectively, and only the second entity had a lousy recall. This may be understood as a problem to the model to define a symptom linked to the complaint itself and a secondary symptom. However, the high number of true positives shows the suitable generalization of the model in these entities since both of them have many possibilities of symptoms. The other entities had few appearances, and the low number of samples with these entities in training resulted in a wrong classification. Along with the few entities, some definitions were a problem to define, as the difference between alleviation and treatment and results and treatment.

Medical history was the section with more numbers of true positives found. The entities related to past illness and past treatments of the patient had some wrong assignments between them in some cases, and this may be explicated on account of many illnesses that had the following treatment, and in some cases, the model defined the entities wrongly. The habit entity had no false positives and nine false negatives. It was perceptible that those false negatives appeared in interviews where the questions about tobacco and alcohol were put differently, showing some lousy generalization. In general, all entities had an excellent precision, being the lowest 81.3% and the biggest 100%, and a good recall for only the entities *Medical History Illness* and *Medical History Treatment*, 82.7% and 90.1% respectively, manifesting the capacity to understand unseen illnesses and treatments.

Family History entities had a good precision (94.5% and 95.4%) and a regular recall (81.3% and 75%) for both entities. Identifying ages and illnesses also had a good precision in early entities, and the same thing happened. The explanation about the lower recall might be the assignments of illnesses to the entity "Medical History Illness", and some given death ages followed an alternative pattern in some cases. Social History had a good f1-score in general. All entities were well defined in the pre-processing, and it was easy for the model to assign the correct entities. The only regular precision belongs to Social History Sleep (77.7%), and in the tests, it was possible to notice that some false positives came from *HPI Course* and *Complaint*

Time.

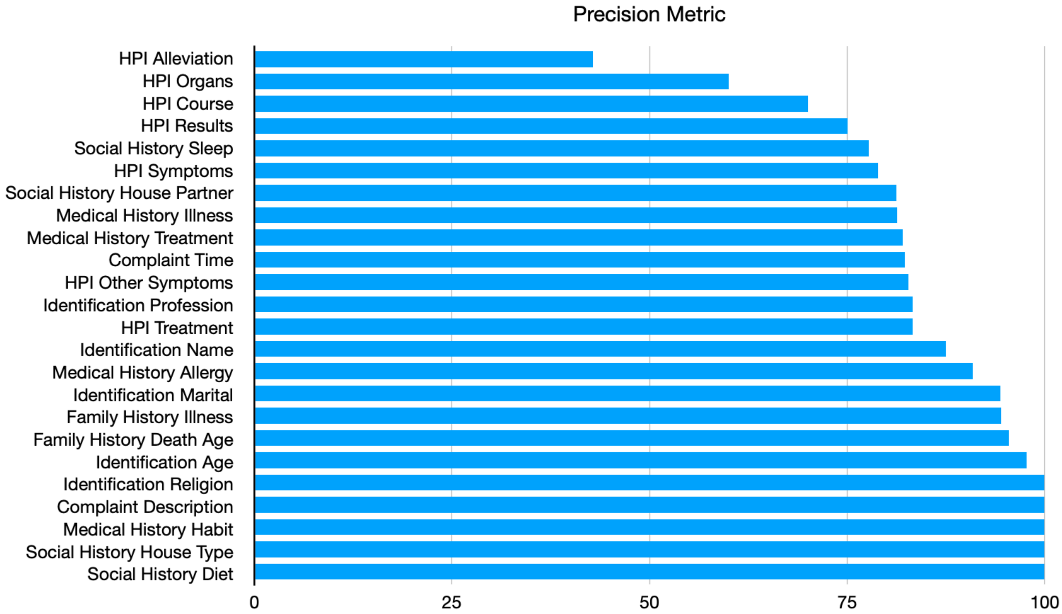
Table 4 – Precision, recall and f1-score by entity and by the model

	Precision	**Recall**	**F1-Score**
Identification Name	87.5 %	82.3 %	84.8 %
Identification Age	97.7 %	100 %	98.8 %
Identification Profession	83.3 %	95.2 %	88.8 %
Identification Religion	100 %	81.8 %	89.9 %
Identification Marital	94.4 %	94.4 %	94.7 %
Complaint Description	100.0 %	94.1 %	96.9 %
Complaint Time	82.3 %	77.7 %	79.9 %
HPI Course	70.0 %	53.8 %	60.8 %
HPI Symptoms	78.9 %	78.9 %	78.9 %
HPI Other Symptoms	82.7 %	58.5 %	68.5 %
HPI Alleviation	42.8 %	42.8 %	42.8 %
HPI Organs	60.0 %	25.0 %	35.2 %
HPI Treatment	83.3 %	45.4 %	58.7 %
HPI Results	75.0 %	50.0 %	60.0 %
Medical History Illness	81.3 %	82.7 %	81.9 %
Medical History Treatment	82.0 %	90.1 %	85.8 %
Medical History Allergy	90.9 %	71.4 %	79.9 %
Medical History Habit	100.0 %	68.9 %	81.5 %
Family History Illness	94.5 %	81.3 %	87.4 %
Family History Death Age	95.4 %	75.0 %	83.9 %
Social History House Partner	81.2 %	100.0 %	89.6 %
Social History House Type	100.0 %	81.2 %	89.6 %
Social History Sleep	77.7 %	87.5 %	82.3 %
Social History Diet	100.0 %	95.4 %	97.6 %
Review of Systems Status	73.7 %	80.3 %	76.8 %
NER Model	85.1 %	87.6 %	86.3 %

Source: Created by the author

Review of Systems is a section with only one entity, the status. In general, it had an excellent f1-score (76.8%). As this is a pretty nonspecific entity, it had a high number of both false

Figure 9 – Graphic containing the precision metric of all entities in ascending order



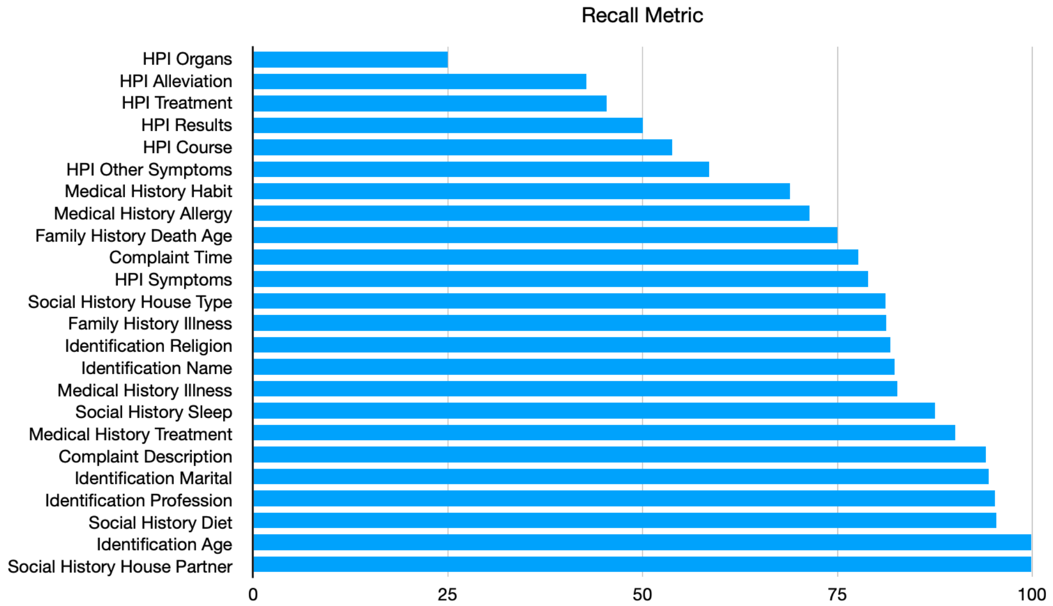
Source: Created by the author

positives and false negatives. The entity's position in the conversation and its question-answer pattern helped the model define the true positives correctly. The NER model had a different calculation of false negatives since it took the whole entity classification. The model had a regular performance (85.1%) and recall (87.6%), resulting in an f1-score of 86.3%. The performance of the sections identification and social history helped the model to have an excellent general performance.

Compared with other works presented in the section of related works, Śniegula, Poniszewska-Marańda e Chomątek (2019) had a comparison work to identify 30 entities using a library that has an option to define the algorithm used. None of these algorithms are implemented by Spacy. The metrics used were the same, but their work also added accuracy. The percentage of the metrics was not shown. The work of Cheng et al. (2019) had a related approach and a similar result. It used a pattern in the questions, specialists to define the labels, and an f1-score of 87% to the complete method. This work is also considered a relaxed way to analyze the results. Akhtyamova et al. (2020) had an f1-score of 90.84% using a corpus with more than 1000 records and only a few entities. Wulff et al. (2020) showed the best results among all papers, with a precision of 97% and a recall of 94%. The results were considerably better than the one proposed here. Lastly, Wen et al. (2021) also used a relaxed way to extract the metrics, and an f1-score of 95.3% was found.

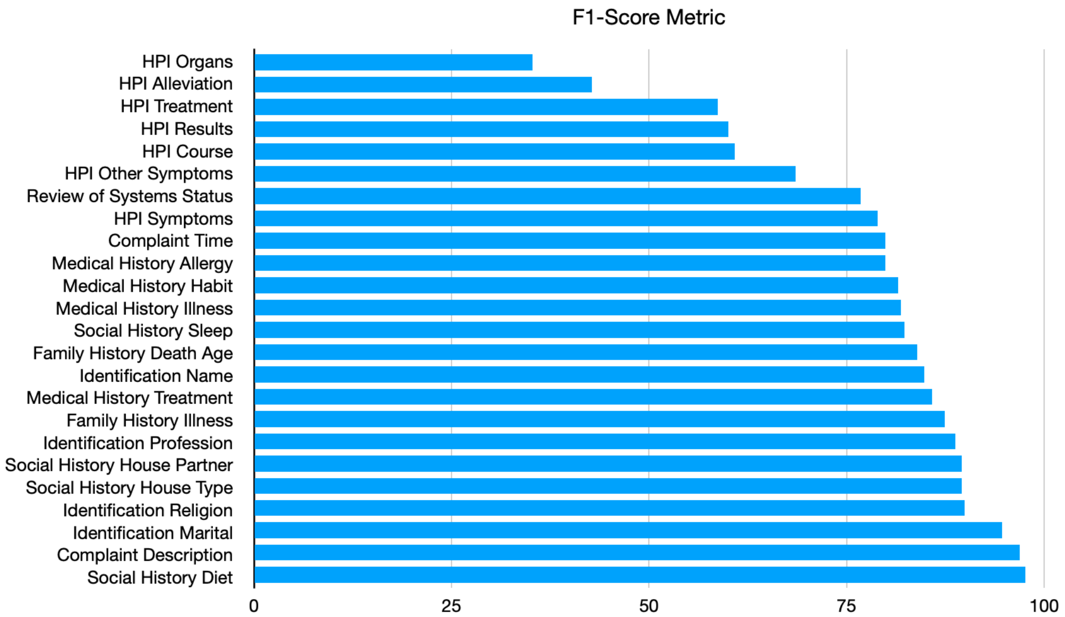
All works found did not show a complete and useful application that could be used by physicians in their interviews. Also, a distinction of this work compared to others was a definition of an output, which has a pattern according to the entities expected in the input.

Figure 10 – Graphic containing the recall metric of all entities in ascending order



Source: Created by the author

Figure 11 – Graphic containing the f1-score metric of all entities in ascending order



Source: Created by the author

6 CONCLUSION

The work shows an implementation of a complete application to recognize the speech of the anamnesis conversation, extract meaningful information using a NER model, and organize the output following a mapping strategy based on the most important information that physicians would like to extract in any medical conversation. The work presented here aimed to be a complete application and define some patterns used in both conversation and output. Twenty-five entities were defined based on the knowledge of experts, and these can serve as a pattern for any anamnesis information extraction.

The limitations found were a better way to analyze the performance of the speech recognition module, which served more like an auxiliary module. Another point is the duality of some entities. The NER model had a worse performance in defining some entities, which caused some misunderstandings by the experts. Another limitation is the hidden settings of the NER model, which is based on the Spacy library. The good side of Spacy is its simplicity, but the bad part is the difficulty of understanding which algorithms are being used and the impossibility of changing them.

A complete application was the immense contribution of this work. None of the articles found had the same approach that this one had. The use of a pre-trained NER model helped develop this project, which could be tested in a few weeks. The application recognizes the entities and creates a JSON following the same pattern recommended in the conduction of the anamnesis. The automatic generation of a digital document was one of the objectives at the beginning of the research, and this was reached. Any physician who uses this application will not be concerned with the patient, which is the best help to the medical community.

The impact of the work against state-of-the-art is that it shows the feasibility to create a complete application to guide all the paths of the anamnesis process. This kind of solution was not found at any other work. Speech recognition and neural networks are areas of study that are becoming very strong. It implies the creation of many frameworks that can easily be used in an integrated way to create many projects related to the automation of anamnesis. Another point of impact is establishing a pattern in both data inputs, the organization of question of the anamnesis interview, and the output document. Both patterns were not found at any work. In addition, a work to recognize entities from anamnesis using speech recognition in the Portuguese was not found.

Future work in this application can better define the entities that need to be identified in the anamnesis. Some of these entities can be unified to have a more straightforward set of anamnesis. For instance, the entities "HPI Symptoms" and "HPI Other Symptoms" had a particular exchange of attribution, affecting the model results. Merging them would be an excellent way to simplify the document and to improve the results. Implementing an EHR specification can be helpful since this is a pattern of modern hospitals worldwide, and this change would help the application be integrated easily. Tests and more features related to the speech recognition module are good options to improve the model. Lastly, an interface connected to the model to start the recording, add manually recorded audios, stop the recording, and edit the document better are needed since it is expected that the physicians will work directly with scripts.

ACKNOWLEDGMENTS

The authors would like to thank all the hospitals that provided the anamnesis for this work to be accomplished. In addition, the authors are also grateful to Martina Schroeder Wissmann, a medicine undergraduate, who contributed to this work by analyzing several documents of anamnesis.

REFERENCES

ACADEMY, D. S. **Deep learning book**. [S.l.: s.n.], 2021. Available at: <https://www.deeplearningbook.com.br/>. Accessed on: oct. 2021.

AGUIRRE, R. R. et al. Electronic health record implementation: a review of resources and tools. **Cureus**, [S.l.], v. 11, n. 9, 2019.

AJAMI, S.; ARAB-CHADEGANI, R. Barriers to implement electronic health records (ehrs). **Materia socio-medica**, [S.l.], v. 25, n. 3, p. 213, 2013.

AKHTYAMOVA, L. et al. Testing contextualized word embeddings to improve ner in spanish clinical case narratives. **IEEE Access**, [S.l.], v. 8, p. 164717–164726, 2020.

CHENG, M. et al. A hybrid method to extract clinical information from chinese electronic medical records. **IEEE Access**, [S.l.], v. 7, p. 70624–70633, 2019.

CZERWINSKA, U. **Named entity recognition - keywords detection from medium articles**. Available at: <https://urszulaczerwinska.github.io/works/egg_ner>. Accessed on: set. 2021.

DERTAT, A. **Applied deep learning - part 1**: artificial neural networks. Available at: <https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6>. Accessed on: jun. 2021.

DICKSON, B. **What are convolutional neural networks (cnn)?** Available at: <https://bdtechtalks.com/2020/01/06/convolutional-neural-networks-cnn-convnets/>. Accessed on: oct. 2021.

ELUYODE, O.; AKOMOLAFE, D. T. Comparative study of biological and artificial neural networks. **European Journal of Applied Engineering and Scientific Research**, [S.l.], v. 2, n. 1, p. 36–46, 2013.

GOYAL, A.; GUPTA, V.; KUMAR, M. Recent named entity recognition and classification techniques: a systematic review. **Computer Science Review**, [S.l.], v. 29, p. 21–43, 2018.

HAMET, P.; TREMBLAY, J. Artificial intelligence in medicine. **Metabolism**, [S.l.], v. 69, p. S36–S40, 2017.

JIANG, F. et al. Artificial intelligence in healthcare: past, present and future. **Stroke and vascular neurology**, [S.l.], v. 2, n. 4, 2017.

KAMATH, U.; LIU, J.; WHITAKER, J. **Deep learning for nlp and speech recognition**. [S.l.]: Springer, 2019. v. 84.

LAI, S. et al. How to generate a good word embedding. **IEEE Intelligent Systems**, [S.l.], v. 31, n. 6, p. 5–14, 2016.

LESLIE, H. openehr-the world's record. **Pulse+ IT**, [S.l.], v. 6, n. 3, p. 50–55, 2007.

LI, S. **Named entity recognition and classification with scikit-learn**. Available at: <https://towardsdatascience.com/named-entity-recognition-and-classification-with-scikit-learn-f05372f07ba2>. Accessed on: jun. 2021.

LICHSTEIN, P. R. The medical interview. **Clinical Methods: The History, Physical, and Laboratory Examinations. 3rd edition**, [S.l.], 1990.

MISHRA, S. **Baffling concept of true positive and true negative**. Available at: <https://towardsdatascience.com/baffling-concept-of-true-positive-and-true-negative-bffbc340f107>. Accessed on:
set. 2021.

OLAH, C. **Understanding lstm networks**. Available at: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>. Accessed on: oct. 2021.

QUIROZ, J. C. et al. Challenges of developing a digital scribe to reduce clinical documentation burden. **NPJ digital medicine**, [S.l.], v. 2, n. 1, p. 1–6, 2019.

SHUNG, K. P. **Accuracy, precision, recall or f1?** Available at: <https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9>. Accessed on: set. 2021.

ŚNIEGULA, A.; PONISZEWSKA-MARAŃDA, A.; CHOMĄTEK, Ł. Study of named entity recognition methods in biomedical field. **Procedia Computer Science**, [S.l.], v. 160, p. 260–265, 2019.

SPACY. **Language processing pipelines**. Available at: <https://spacy.io/usage/processing-pipelines>. Accessed on: set. 2021.

TERRY-JACK, M. **Nlp**: pretrained named entity recognition (ner). Available at: <https://medium.com/@b.terryjack/nlp-pretrained-named-entity-recognition-7caa5cd28d7b>. Accessed on: set. 2021.

VERMA, Y. **Complete guide to bidirectional lstm (with python codes)**. Available at: <https://analyticsindiamag.com/complete-guide-to-bidirectional-lstm-with-python-codes/>. Accessed on: oct. 2021.

WEN, C. et al. Medical named entity recognition from un-labelled medical records based on pre-trained language models and domain dictionary. **Data Intelligence**, [S.l.], p. 1–13, 2021.

WILAMOWSKI, B. M. Neural network architectures and learning algorithms. **IEEE Industrial Electronics Magazine**, [S.l.], v. 3, n. 4, p. 56–63, 2009.

WULFF, A. et al. Designing an openehr-based pipeline for extracting and standardizing unstructured clinical data using natural language processing. **Methods of information in medicine**, [S.l.], v. 59, n. Suppl 2, p. e64, 2020.