Master Thesis

Extracting Relative Temporal Relation Labels of ICF Functioning Statuses in Dutch Medical Notes

Meruyert Nurberdikhanova

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by: Prof. Dr. Piek Vossen
2^{nd} reader: Dr. Pia Sommerauer

Submitted: August 15, 2023

Abstract

Temporal relation identification is an important task in clinical applications of NLP. It is one of the prerequisites for the recovery timeline construction of patient's health. In this thesis two traditional Machine Learning approaches are compared on their predictions of temporal relation labels in Dutch medical notes: a feature-engineered Support Vector Machines model and a fine-tuned medical domain Large Language Model, MedRoBERTa.nl. The models are trained on medical data of annotated ICF functioning statuses from Amsterdam University Medical Center, as part of an ongoing A-PROOF project. The fine-tuned MedRoBERTa.nl has performed better than the SVM model by a small margin, with a macro f1-score of *.77*. Although, it is worth noting that the application of this model would be limited to the specific language use of AUMC medical professionals, whereas generalisability of the model is still under question. Future research should be directed towards overcoming constraints of the experiments described in this thesis and construction of an absolute temporal relation identification classifier.

Declaration of Authorship

I, Meruyert Nurberdikhanova, declare that this thesis, titled *Extracting Relative Temporal Relation Labels of ICF Functioning Statuses in Dutch Medical Notes* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: August 15, 2023

Signed: Meruyert Nurberdikhanova

Acknowledgments

I would like to thank my supervisor, Prof. Dr. Piek Vossen, for continuous support throughout the development of this thesis, especially despite all the delays and obstacles that we have found along the way. I would also like to extend my gratitude towards Edwin Geleijn, the lead from the A-PROOF project team, for answering all of the questions I had and being present both on our supervision meetings and throughout all the email exchanges.

My everlasting gratitude goes to Bertjan Busser for constant technical support throughout all the frustrating issues we have encountered with using AUMC secure remote servers. This thesis would never have been finished without his help. I would like to thank Cecilia Kuan and Cecilia Schramm for going through this internship together with me, for complaints, advice, tears, and laughter. I feel we have become ever so close because of this.

I would also like to thank all of the people that have been mentally and physically supporting me in my writing and coding process. Thank you, Nemo, Marijke, Seth, Merlijn, I love you and I appreciate everything you have done for me. I also feel thankful for my friends that have been there for me and who I love dearly, Kavi, Tiah, Aina, Sofa, and Nami. I want to also thank those who have helped me find this love for linguistics during my bachelor's degree, thank you, Gaetano and Serra, you are wonderful people and I hope we get to sit together and discuss morphosyntax again soon.

I would finally want to thank friends that I have made in this master and who have gone through the same processes as I did to get here. I hope this is only the beginning of our blooming friendship, here is to many more years to come, Sofia, Swa, and Sal.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Motivation

This thesis is an extension of an ongoing research led by A-PROOF team (Automated Prediction of post-COVID RecOvery Of Functioning) in using Natural Language Processing (NLP) to monitor patients' functioning on a larger scale in Amsterdam University Medical Center (AUMC) (The A-PROOF Project, 2023). To determine functioning of the patients, Kim et al. (2022) used International Classification of Functioning, Disability, and Health (ICF) to determine nine major categories of functioning statuses in clinical notes as well as the qualitative levels of functioning within those categories. The categories used were: energy level (b1300), attention (b140), emotional functioning (b152), respiration (b440), exercise tolerance (b455), weight maintenance (b530), walking (d450), eating (d550), work and employment (d840-d859); which all can be found in the ICF catalogue (World Health Organization, 2023). The ICF is seen as a good way to standardise measurement of human health across different institutions and countries. Maritz et al. (2017) found that ICF guidelines' general comprehensibility and application to free-text data are its main benefits for quality improvement in rehabilitation and other medical frameworks.

The goal of the A-PROOF project is to be able to map out functioning patterns of patients at the hospital and determine commonalities in recovery patterns of certain diseases. This way, this project can help ease the rehabilitation process of the patients and prevent disability development by determining larger-scale similarities in symptoms, progression of functioning, and other variables involved. So far, for A-PROOF project Verkijk and Vossen (2021) have trained their own medical domain RoBERTa model on the millions of clinical notes available at the AUMC - MedRoBERTa.nl - and Kim et al. (2022) have fine-tuned this model for ICF categories and levels classification.

However, in medical notes, ICF functioning is frequently mentioned in relation to past functioning of the patient or expected future functioning, estimated by the medical professionals. This has brought out another challenge in reconstruction of patients' recovery patterns: notes with their respective ICF categories and levels labels are placed on the timeline of the patient by the note creation date, but if note references a non-current functioning of the patient, the daily functioning progression along the recovery timeline would be skewed (Kim, 2021). Thus, automatically detecting whether sentences are referring to current, past, or expected functioning of the patient could help with accuracy of reporting this information for recovery patterns.

1.2 Goal and research questions

The goal of this thesis is then to explore machine learning approaches that could help to distinguish the temporal aspect of patients' functioning explicitly indicated in their medical notes. My main research question is as follows: What is the more suitable approach to creation of relative time identification classifier in the medical domain? And therefore I would also want to elaborate on the following subquestions:

1. How do feature-engineered Support Vector Machine (SVM) model and fine-tuned MedRoBERTa.nl compare on prediction of relative time labels in sentences taken from Dutch medical notes?

2. Are there qualitative differences between sentences under labels "past", "now", and "future"?

3. How can relative time labels be mapped to absolute time labels?

4. Are there qualitative differences between the same time labels in different ICF functioning categories?

To answer these questions, I am going to build two types of classifiers, Support Vector Machine and fine-tuned MedRoBERTa.nl, and compare their evaluation metrics and errors with each other. I will also explore some corpus statistics between the temporal subsets of my data and look into the next steps of applying the end-product model in the larger A-PROOF project.

1.3 Outline

In this thesis I will first delve into background information on temporality in clinical data in chapter 2. I will look into the applications of NLP in the medical domain and I will also look into the research previously done specifically on temporal information extraction subtasks. Then, I will operationalise the task at hand and discuss the methodological settings of my experiments (the data, the models, and the features) in chapter 3. There, I will also describe some of the qualitative characteristics of the data, as well as try to ground my hypotheses about linguistic features needed for this task in the statistics of the training data. Afterwards, I will report the results and describe the error patterns of my systems in chapter 4. I will discuss the results and how they relate to my research questions and further body of research in chapter 5 and I will definitively answer those questions in chapter 6. Additionally, I will provide some more information on the constraints of the current research, as well as the points of improvement and further work on this topic.

Chapter 2

Background and Related Works

In this chapter I will discuss in detail the application of temporal information extraction in clinical data. I first look into the different applications of clinical data in NLP and how they benefit the medical professionals and their patients. Then, I explore how temporal information has been extracted and used in NLP and stemming from that different methods and challenges of getting temporal information out of the medical data.

2.1 Clinical Application of NLP

Clinical domain application of NLP became more possible as the move to Electronic Health Record (EHR) databases became more acceptable in the medical centers. Nowadays, most of the patient records, medical notes, background information, disease history, and other relevant information can be found in the EHR systems of the hospitals. These records mainly consist of unstructured textual data created at an enormous rate of daily additions to patient files. This unstructured data is then given to NLP researchers who deal with automation of analysis of such large volumes of data and with extraction of some structured information out of it (Kreimeyer et al., 2017).

However there is a number of challenges associated with clinical data besides its accelerated rate of collection that distinguish it from other text-based data. Specifically, clinical data is highly sensitive: it cannot be put in open access without concerns for the leaks in privacy (Suster et al., 2017). A lot of personal information of the patients can still be traced down even after anonymisation of the data and the models. Thus, researchers in clinical NLP are often strained either by the small sample size of openly available clinical data or by the heavy sampling bias of being limited to one institution's clinical records (Suster et al., 2017). There are no unified medical databases that link entirety of hospitals and other medical institutions within the same country as of yet, since the laws and security concerns around clinical data and under whose supervision should they be unified have not been worked through.

Clinical texts are characterised by domain-specific language, using short-hand writing, and irregular syntactic structures among other qualitative difference from other forms of linguistic data (Kreimeyer et al., 2017). And in some cases, clinical data can be overly explicit and have a technocratic style due to legal implications of reporting symptoms, actions, and stating diagnoses to patients. These characteristics distinguish clinical data from other textual data used in NLP, which makes it more challenging to apply existing systems and observations about free-text data to this domain. Per-

forming a thorough corpus analysis of the given data may help researchers familiarise themselves with these differences. However, often times there is no room in planning of the research for such lengthy process.

Additionally, although we refer to the clinical data, especially in EHRs as clinical notes or records, this data by no means is uniform in its style and presentation (Verkijk and Vossen, 2021). Clinical notes come from a variety of sources, such as general practitioners, psychologists, physiotherapists, nurses, and other specialists, as well as in a variety of formats like quick visiting notes, primary care notes, treatment plans, prescription notes, and many more. Whether differences between notes will be crucial to the results of your research would depend on the goal and scope of the clinical NLP application in development.

Most clinical NLP applications are primarily concerned with pattern recognition within free-text data. With ever growing size of unstructured data available to the hospitals, it is not surprising that finding relations between notes of the same subdomain on a larger scale is something that medical professionals are seeking. For example, finding common symptoms and comparing their development across subset of patients with the same diagnosis. These types of research can help increase efficiency and generalisability of dealing with clinical records for medical professionals.

Some of the common applications of NLP in clinical domain include information extraction, creation of domain-specific models, clinical Named Entity Recognition (NER), temporal information extraction (Névéol et al., 2018). These applications can be further hindered if you are trying to develop tools in languages other than English, as the research in NLP is mainly relying on English language and many of the tools available for it are not well developed yet for other target languages. However, clinical research is of great importance for all nations and language communities, which is why it is so crucial to continue developing natural language understanding of all human languages.

Similarly, when the development of application in clinical NLP goes on, problems arise from the lack of annotations for the given task. And when there are annotations and annotators of medical background available, often they do not have the capacity to produce a sizeable set of annotations needed for training of, for example, Large Language Models (Styler IV et al., 2014). These annotations thus are both valuable and costly, requiring a considerable amount of confidence and determination from the medical team to continue funding such projects. But if the research projects persevere this stage of development, important observations have to be made that would benefit the clinical NLP community as a whole.

2.2 Temporal Information Extraction

One of the applications in clinical NLP that is still quite challenging is Temporal Information Extraction. This is a task that has been investigated in NLP for quite some time, starting off with biographical and historical data (van de Camp and Christiansen, 2013). Temporal information extraction is concerned with labelling temporal expressions within unstructured texts and finding temporal links between different events, for example in biographical texts that would mean anchoring events like attending an educational institution to a certain span of years in the person's lifetime (Verhagen et al., 2009). However, temporal expressions can be challenging to find in the text, especially considering that they are not always explicitly stated in a sentence, where a reader would be able to connect it with contextual information given in the previous

sentences but a machine may not be able to successfully pick up on the coreference of the same entity from previous sentences (van de Camp and Christiansen, 2013). Moreover, temporal markers in discourse can vary greatly, as it can be just the use of past tense in a sentence or explicit statements about the date or time of an event, as well as the relative temporal markers (Leeuwenberg and Moens, 2018).

The definition of an event itself is broad and can include a variety of actions, diagnoses, medications, and others when it comes to the medical field (Styler IV et al., 2014). Similarly, linking events in a timeline also requires the events to be all properly labelled and explicitly stated in text, limiting the scope of applicability of the temporal information extraction systems. This led to development of shared tasks that focussed specifically on temporal information extraction - TempEval (Verhagen et al., 2009, 2010; UzZaman et al., 2013).

In previous research it was common to base off temporal information extraction on Allen's algebra: a set of logical constraints to account for both temporal points (dates) and intervals (relative time expressions) explicitly stated in text (Allen, 1983). But with the knowledge of implicit temporal expressions, the methods have shifted to development of linguistic constraint rules for finding these temporal expressions (van de Camp and Christiansen, 2013). Similarly, with the development of Large Language Models (LLMs), more and more research has shifted its focus from rule-based and feature-explicit systems to fine-tuning LLMs to their task (Devlin et al., 2018). But evaluation of natural language understanding by the machines and specifically the task that LLMs are made for is still a big question in the research community. Black-box Machine Learning systems often lack interpretability and with the rise in their popularity and deployment for every textual task out there, we have seen also a rise in criticism of whether they actually work better than feature-explicit systems (Belinkov and Glass, 2019). Thus, naturally came a question of whether a feature-explicit model would work as well as a LLM model for the particular goal of this thesis.

Due to complexity of temporal information extraction and desire of researchers to get as much structured information out of text as possible, it is often divided into subtasks: event extraction, temporal expression extraction, temporal links between events, temporal expressions, and both. Temporal links can also be further anchored to document creation time (DCT), which then categorises these links into references to time before the DCT, overlapping the DCT, after the DCT or before and overlapping, as based on Allen's algebra (Gumiel et al., 2021). This can create relative timelines of events, which then further can be expanded to absolute timelines by attempting to probabilistically determine bounds of event durations (Leeuwenberg and Moens, 2020). This complexity of steps is ultimately where with gaps in information, accurate classification can be lost.

In terms of clinical data, temporal information extraction can be useful to monitor progression of diseases, symptoms, treatments, recovery of patients, and other uniquely clinical events that could be further summarised and reviewed by medical professionals in patient's history (Styler IV et al., 2014). And with the data stored in EHR databases, you are guaranteed to have DCT available somewhere in the metadata of each clinical note. However, as aforementioned, there are many challenges in processing clinical data, which also affect temporal information extraction. Specifically, the inconsistent stylistics of the documents from different medical professionals and unique mix of technocratic and short-hand medical slang writing can hinder success rates of temporal information extraction classifiers. This is mainly due to excruciating amount

of details needed for the pipeline of subtasks in temporal information extraction which lead to a large amount of time used on annotations for them or on realisation that many of these details are not explicitly stated in text. I will discuss these issues later in chapter 5 in greater detail.

In summary, there is a variety of clinical NLP tasks that can be done and used for the benefit of patients and the medical professionals. Temporal information extraction task is one of the important and quite complicated such tasks in clinical NLP. The pipeline of the multiple, very precise subtasks in temporal information extraction makes it quite a challenging task to train Machine Learning classifiers for.

Chapter 3

Methodology

3.1 Task

In this thesis, I focus on the sentence-level classification of relative temporal relation between the ICF functioning event and document creation time. The labels are simplified to *past* as in past functioning status of the patient before the DCT, *now* as in functioning status on DCT of the note writing, and *future* as in expected future functioning status after DCT. While usually temporal relations or TLinks are defined between explicitly labelled events and temporal expressions in the text, this was not possible with this annotated data, as the events - ICF functioning categories - were not anchored to specific tokens in the sentences for past and expected functioning, instead an ICF category label is applied to the entire sentence (Kim, 2021). Similarly, no annotation were made regarding explicit temporal expressions used in these sentences. Therefore, we are performing a text classification task: where the unit of model classification is a sentence, instead of a token.

3.2 Data

For this project, I mainly utilised data used by Kim et al. (2022) for ICF functioning category and level classification. In this section I will focus on description of the subset of that data used in this thesis, but for further information on the overarching statistics please consult the paper by Kim et al. (2022). It is also important to promote usage of data statements in relation to ethical concerns in the field of NLP as described by Bender and Friedman (2018), hence the data statement composed for the current dataset can be found in Appendix A. This data statement was filled out by me, based on the information available through the discussions with the leads of the A-PROOF project as well as the papers so far published in relation to the project.

3.2.1 Annotation

Out of then available 10 million notes at AUMC, Kim et al. (2022) annotated around 6.000 notes on the sentence-level (around 286.000 sentences) with ICF functioning categories and levels for the functioning of the patient on the day of writing the note. The token that explicitly referred to the ICF category would be labelled with that category, whereas the phrase that explicitly referred to the level of functioning would be labelled with the level annotation - between 0 and 5 or between 0 and 4. However, if the

sentence discussed ICF functioning before the DCT, the entire sentence was annotated as *background* sentence, while if it discussed expected future ICF functioning, it would be annotated as *target* sentence.

Kim (2021) has noted that in retrospect, all of the sentences discussing ICF functioning should have had explicit category and level annotations regardless of their timeline. In the current research, this brings an issue, because for two subsets of the data, *past* and *future*, there are no tokens annotated with gold explicit events that we are trying to map out on the timeline of the patients' recovery patterns. Since our goal is to distinguish these non-current ICF category references from the current ones, I have decided to stick with the sentence-level annotations of ICF categories. This is also why it is important that the end classifier would have as high of a performance on these minority labels as possible, as they are at the biggest disadvantage in identification of the relative temporal relation.

3.2.2 Descriptive statistics

In Table 3.1, you can see that the utilised data (Jenia's column) consists of 242.291 sentences for training set, 21.742 sentences for development set, and 22.082 sentences for testing set. To create the training, development, test data split for this thesis I have used gold labels of ICF functioning categories and labels used for marking past ICF functioning status or expected future ICF functioning status as my filters of Kim et al. (2022)'s data. As described by the authors, frequency of these labels is quite low within the natural distribution of electronic health records (Kim et al., 2022). Medical professionals would mention one of the nine functioning statuses of their patients only 5,15 % of the time in training set, 7,18 % in development set, and 7,06 % in test set in relation to their current health. Whereas, the past functioning has been described only in 1,73 % of the sentences of the training set, 2,41 % of the development set, and 2,37 % of the test set. The expected future functioning was the least prominent in the notes, where it made up only 1,12 % of the training set, 1,55 % of the development set, and 1,52 % of the test set.

	Jenia	past	past %	now	now %	future	future %
train	242.291	4.197	1,73	12.485	5,15	2.708	1,12
dev	21.742	525	2,41	1.561	7,18	337	1,55
test	22.082	524	2,37	1.560	7,06	336	1,52
total	286.115	5.246	1,83	15.606	5,46	3.381	1,18

Table 3.1: Frequency of temporal subsets in Jenia's data (by sentence).

Overall there are 5.246 sentences labelled as past functioning, 15.606 sentences labelled as current functioning, and 3.381 sentences labelled as future functioning. To preserve the natural distributions of these temporal labels in the data, I have split each temporal subset into training, development, and test sets by the 80/10/10 ratio and recombined them in a shuffled order. Then, for my SVM system I needed to do some preprocessing of the sentences, where some duplicates were revealed and removed (discussed at length in subsection 3.2.3). This way, as you can see in Table 3.2, my final split consisted of 24.233 sentences: 19.390 sentences in training set, 2.423 sentences in development set, and 2.420 sentences in test set. I decided to only keep the positive examples to avoid working with majority label of my classifiers being a None label.

	train	dev	test
past	4.573	337	336
now	12.485	1561	1560
future	2332	525	524
total	19.390	2.423	2.420

Table 3.2: Frequency of temporal subsets in my train/dev/test split.

Within the data split I made, there were 7.398 medical notes used overall: 4.229 in training set, 1.577 in development set, and 1.592 in test set. These notes were written about 4.581 patients: 2.547 patients are in training set, 1.012 patients in development set, and 1.022 patients in test set (Table 3.3).

	notes	patients
train	4.229	2.547
dev	1.577	1.012
test	1.592	1.022
total	7.398	4.581

Table 3.3: Frequency of unique notes and patients (by sentence).

Similarly, you can see in Table 3.4 the data consisted of 3.760 sentences written in the year 2017, 3.464 sentences in the year 2018, and 17.009 sentences in the year 2020. And in Table 3.5, you see that 14.289 sentences came from Amsterdam UMC location and 9.944 sentences came from VUMC location.

	2017	2018	2020
train	2.981	2.794	13.615
dev	381	354	1.688
test	398	316	1.706
total	3.760	3.464	17.009

Table 3.4: Frequency of years in the data (by sentence).

	AMC	VUMC
train	11.482	7.908
dev	1.418	1.005
test	1.389	1.031
total	14.289	9.944

Table 3.5: Frequency of institutions in the data (by sentence).

3.2.3 Preprocessing

As part of feature engineering and feature extraction for the SVM model, I have changed representation of my data from .TSV files where each sentence occupied one line to CoNNL format with each token occupying one line. I did this to be able to engineer features on token-level without confusions (discussed in section 3.5.1). However, while

extracting a note length feature that requires looking through raw files with medical notes' metadata, I have found out that there are duplicate ID numbers for some notes dating to year 2020. This unfortunately left me with duplicated sentences as well as a decision on which one of the duplicate ID notes to keep.

I had a discussion with the medical team lead, Edwin Geleijn, who speculated that due to high volumes of patient intake in 2020 some notes have been updated as the day and events have progressed. Thus, virtually overlapping in text notes were saved in the EHR system under the same ID number. On further investigation, some of them were updated on the same date and some on a later date. I decided to keep either the notes that were updated on the later date if duplicate IDs had different dates or the notes that were longer if duplicate IDs had the same document creation date. Therefore, table 3.2 reflects these preprocessing changes in count. Interestingly, training set had 56 notes that had a duplicate note ID, dev set had 25 duplicate notes, and test set had 26 duplicate notes.

3.3 Evaluation metrics

I will mainly base my comparison of the systems on standard evaluation metrics used in NLP: precision, recall, and f1-score. I look at these metrics per temporal label, as well as the macro f1-score of the whole system. Additionally, I am mindful of the fact that *now* is a majority label in the dataset, which makes me inclined to favour a system that has better scores for the minority labels - *past* and *future* ICF functioning - as it is a baseline expectation that systems should be able to predict a majority category well.

3.4 Models

To create a robust automatic relative time classifier of ICF functioning, I compare two major Machine Learning approaches: feature-explicit model and Large Language Model fine-tuned for the task. These types of systems are often seen as a contrast between the expert thought out features fed into a system and the machine powered relation detection on the large amounts of training data. I will specifically compare Support Vector Machine system and fine-tuned medical Dutch RoBERTa system. Thus, my experiments are divided into two major branches: feature creation, combination, and testing for the SVM model, as well as fine-tuning of MedRoBERTa.nl model. For both branches, after experimentations on development set, I will test final two models on test set and evaluate their advantages and pitfalls by careful error analysis.

3.4.1 SVM

Among feature-explicit systems, Support Vector Machine models are a popular choice for their ability to work with correlated features at least until the rise in popularity of LLMs. In NLP field it is unavoidable to have correlated features due to the complex nature of language. In temporal information extraction task, SVMs are commonly used for temporal relations identification and temporal expression identification (Lin et al., 2016; Gumiel et al., 2021; Roberts et al., 2013). Alternatively, I could have tried out a Conditional Random Fields (CRF) model - another popular feature-explicit choice for the task, but due to software developments in the past decade CRF models are less

accessible and compatible with other packages used with models like SVM or Logistic Regression.

For my experiments with SVM model, I have looked through the literature on features used in temporal information extraction task and parameters for the model set up. Commonly used features and my choices for this system are described in section 3.5.1. And when it comes to the parameters, as commonly a problem in published papers, no specifications of parameters for feature-explicit models - the way they are done for LLMs - have been stated in any papers I have looked through.

Thus, I have tried to test out several kernel types, linear and Gaussian, as well as various feature representations, combining one-hot-encodings, weighted representation of tokens, and numeric entries. At the end, I have narrowed it down to 4 experiments set up with linear kernel that differed in their feature combinations: an experiment for morphological features, an experiment for discourse features, an experiment for temporal features, and a final experiment picking the most suitable features from the previous three experiments. The end system is a linear SVM model using eight features, not counting the tokens, from the experiment number four.

3.4.2 MedRoBERTa.nl

As described by Devlin et al. (2018), the idea is that LLMs attempt to grasp on the linguistic peculiarities of the texts they are trained on, generalise for a specific domain. These models would make sense to use for domains like clinical where the language qualitatively differs from the general text-based communications. However, it is also important to keep in mind what Belinkov and Glass (2019) and others have pointed out: how LLMs tend to get into pitfalls of cutting corners and we still have no idea what do they actually do despite their impressive performances.

For the LLMs in clinical domain, Verkijk and Vossen (2021) showed that pretrained on medical Dutch notes model is better suited for various NLP tasks using clinical texts compared to more general Dutch LLMs like BERTje and RobBERT. Verkijk and Vossen (2021)'s model was then fine-tuned for ICF category and label identification tasks by Kim et al. (2022).

It is only logical to then try fine-tuning the same model on the task of temporal relation identification of the very same ICF categories. Additionally, since the temporal relation identification model would be used within the AUMC complementary to Kim et al. (2022)'s model, it makes sense to use MedRoBERTa.nl pretrained on the AUMC medical notes.

To fine-tune MedRoBERTa.nl for temporal relation identification task I am using simpletransformers package for compatibility with Kim et al. (2022)'s system made with the same software (Rajapakse, 2019). Just like Kim et al. (2022), I am using default parameters for the RoBERTa model: batch size of 8, learning rate of 4e-5, and 1 epoch. I have decided to cautiously keep fine-tuning at 1 epoch, as it is easy to overfit the model on the small dataset like ours.

3.5 Features

Among feature-explicit systems for temporal information extraction there is a variety of features that researchers have successfully implemented for all the subtasks. As mentioned in chapter 2, the main issue with building a model for temporal relation ex-

traction is that this subtask involves already identified temporal events and expressions. However, we do not have annotations for the temporal expressions and the annotations for the events - ICF categories in this case - are not attached to a specific token.

Feature-explicit models frequently deployed event-centered features for temporal relation identification subtask: event polarity, event part-of-speech tags and tokens, presence of the event tokens in Unified Medical Language System (UMLS), event position in the medical note (Gumiel et al., 2021; Roberts et al., 2013; UzZaman and Allen, 2010). These event-centered features, unfortunately, cannot be used in our case. However, we can implement other types of features - morphological and temporal. The morphological features are used frequently for all subtasks of temporal information extraction: part of speech tags and morphosyntactic information of events and temporal expressions in particular for temporal relation identification (Gumiel et al., 2021; UzZaman et al., 2013). As, again, annotations for events and temporal expressions are not available on token level of the data, I have traded off to use the morphological features for all the tokens of the sentence. When it comes to temporal features, in previous literature temporal expressions' positions in the text as well as simple indicators such as verb tense have been used by researchers (Gumiel et al., 2021). Lack of gold temporal expression annotations could be compensated by another predictor of temporal expressions, such as temporal Named Entity Recognition (NER) system. Whereas verbs and their tenses can be engineered from the morphological features.

Thus, keeping the limitations of our current dataset in mind, I have focussed mainly on three broad categories of features for SVM classifier: morphological, temporal, and broad discourse features. The morphological features are Universal Part of Speech tags (UPOS), extensive Part of Speech tags (XPOS), and universal syntactic dependency tags (deps) (Petrov et al., 2012; Van Noord et al., 2013; Nivre et al., 2020). The temporal features consist of temporal labels of NER classifier, presence of verbs and their tenses in the sentence. And the discourse features are medical note's length and sentence position in the note. Sentence position in the note is inspired by the feature of event position in the note (Gumiel et al., 2021). Whereas note length feature is a numerical feature that was already available: it may be useful to distinguish "now" label from the other two, based on how much details and for how long does the medical note stretch.

In the next subsections I will describe the technical implementation of the chosen features for SVM classifier and some of the distributional statistics of these features in the training set of the data. This way I can try to ground my theoretical choices for features within the given data to see if I can expect certain features to work better than others for the temporal relation identification task.

3.5.1 Feature engineering

For SVM implementation, as discussed earlier, I have created four experiments: morphological, temporal, discourse, and a combined experiment. Features used for the morphological experiment are numbered 1 through 3 below, features for temporal experiment are 4 through 6, and features for discourse experiment are 7 and 8. The combined experiment ended up using all of the above features, as you will see in chapter 4, since none of the features significantly worsened the performance of the classifier.

As described in section 3.2.3, I have converted the dataset into CoNLL format in order to get all the features. SVM classifier just as MedRoBERTa.nl was set up to take a sentence as their input unit and give a single label for the entire sentence as their

output. Therefore, while features were engineered and collected on the token level, majority of them were combined into lists on sentence level to be vectorised and used as input for the classifier. For example, if a hypothetical sentence looks like *Patient experiences shortness of breath, fatigue, and low mood*, then it's UPOS feature for the sentence would be a list of UPOS tags of each token, such as [*NOUN, VERB, NOUN, ADP, NOUN, PUNCT, NOUN, PUNCT, CCONJ, ADJ, NOUN, PUNCT*]. The same list representation goes for all morphological and temporal features, where order of features in the sentence is deemed important. However, for the discourse features only one instance per sentence was added to feature vector of the sentence. This is because note length and position of the sentence in the note do not change throughout tokens in the sentence.

The features I have used in my SVM experiments are as follows:

0. **Tokens** I have used tokens of the sentence without modifications made to them. But when it came to vector representations, there were several issues stemming from the qualitative differences of clinical domain language from general language. I will focus more on those in chapter 5 but, in short, one-hot-encodings did not work for this feature. Instead, tokens were represented using Term Frequency - Inverse Document Frequency (TF-IDF) weighting supplied by TF-IDF vectoriser from scikit-learn package (Pedregosa et al., 2011). In this case, I used it as terms being tokens and sentences being documents in the equation. Thus, the weights of tokens were relative to the inverse frequency of them in all sentences of the dataset. A different approach to TF-IDF weighting is also discussed in chapter 5. Additionally, due to lowercase normalisation of TF-IDF vectoriser, my token representation dimensions reduced from 19.343 to 13.307.

1. **Part-of-speech tags of the tokens** For each token in the sentence I have extracted universal part of speech tags that supposedly are abstract enough to distinguish patterns of their occurrences in sentences (Petrov et al., 2012). UPOS tagset includes 17 labels of which the Dutch model uses 16 (with the exception of "particle" or PART), providing only a small number of dimensional increase for our vector representations. I used spaCy package's Dutch model[1] to extract these UPOS tags for each token combined into a sentence list, which then I converted using a dictionary vectoriser from scikit-learn to sparse representations (Honnibal et al., 2023; Pedregosa et al., 2011).

2. **Extensive part-of-speech tags** Fine-grained part-of-speech tags provide some distinguishing aspects for each tag that are linguistically unique to the Dutch language. These tags were based on work made by Van Noord et al. (2013) for syntactic annotations of written Dutch language. The unique thing about these Dutch tags is that they combine what XPOS tags and the morphosyntactic information labels separately provide in English models. This provides additional level of detail, that can help resolve some syntactic ambiguity of tokens in the sentence (Van Eynde, 2005). For example, for *read* in *I read a book*, the XPOS tag in English would be *VBZ* and the morphological information would be *VerbForm=Fin|Mood=Ind|Tense=Pres*. In an equivalent Dutch sentence *Ik lees een boek*, *lees* would be tagged as *WW|pv|tgw*, where *WW* is *werkwoord* or verb, *pv*

[1] nl_core_sm-3.5.0: `https://github.com/explosion/spacy-models/releases/tag/nl_core_news_sm-3.5.0`

is *persoonsvoorm* or finite verb form, and *tgw* is *tegenwoordig* or present tense (Van Eynde, 2005). Thus, there is a larger collection of labels for these POS tags (160 tags) than for, e.g., Penn TreeBank POS tags (Marcus et al., 1993). Just as with UPOS tags, I used the Dutch model of spaCy and a dictionary vectoriser of scikit-learn to get the feature representations.

3. **Syntactic dependency tags of the tokens** Syntactic dependencies can provide more direct connections between tokens in the sentence. Using spaCy dependency parser, I have obtained universal dependency tags of tokens (77 tags), which were again vectorised on a sentence-level using scikit-learn (Nivre et al., 2020). Originally, for this feature, I have tried to encode it as a combined head of the token and its dependency tag feature. So instead of just *nsubj* for *Ik* in *Ik lees een boek*, it would be *lees:nsubj*. However, even with the small size of the dataset at hand, this resulted in 57.951 dimensions of the sparse vector representations. Thus, I have decided to use dependency tags by themselves.

4. **Tense of the verb** Verb tense is often one of the most simple temporal indications in text. Thus, I constructed a simple tense feature based on the fine-grained POS tags: where finite present tense verbs, *WW |pv |tgw*, would constitute as *present* tense feature, finite past tense verbs, *WW|pv|verl*, as *past* tense feature, and infinitive modal verb constructions, *WW|inf|vrij*, as weakly associated with *future* tense feature. For non-verbs and verbs that do not fall into these categories, a simple *None* category is assigned. The end sentence features would look like [*None, present, None, None*] for *Ik lees een boek*. This then is encoded as sparse vector representations, adding 4 more dimensions.

5. **Verb** To try to catch possible other correlations with time that verbal structures can provide in text, I have also used this verb feature. Simply, it encodes any of 19 verbal XPOS tags present in the dataset as they are named and assigns *None* category to other tokens. So, our example sentence *Ik lees een boek* would have [*None, WW|pv|tgw, None, None*] as its verb feature. And again, this feature is transformed via dictionary vectoriser from scikit-learn (Pedregosa et al., 2011).

6. **Temporal expressions** With lack of gold temporal expressions, I have decided to implement the NER classifier available in spaCy package for the Dutch language. Out of 18 NER labels, two are concerned with temporal information: *DATE* AND *TIME*. Tag *DATE* refers to absolute and relative mentions of dates, whereas tag *TIME* refers to mentions of time that are less than a day long (Weischedel et al., 2013). These two tags were used as they are, whereas I ignored the other 16 tags for this feature and replaced them and tokens that are not classified as NER with label *None*. For example, in a sentence *Expected improved breathing by next appointment* this feature would be represented as [*None, None, None, None, TIME, None*]. And this feature is also encoded as a sparse vector with additional three dimensions.

7. **Sentence position in the note** Relative sentence position in the medical note could indicate whether the patient's functioning is discussed in the past, current state, or expected future state. This is mainly due to discourse structures of written text: patient history would likely be at the beginning of the note, with the current ailments at the beginning and the main core of the text, whereas expected

future recovery and issues would be listened at the end of the text. Based on this hypothesis, I decided to represent sentence position in note quartiles: whether the sentence is in the first (*Q1*), second (*Q2*), third (*Q3*), or fourth quarter of the note (*Q4*). To calculate these quartiles I consulted metadata, as discussed in section 3.2.3, where full note texts and sentence numbers are available. Then, using a sentencizer from spaCy I counted the number of sentences in the notes I needed - this is our note length. Using sentence number and dividing it by the note length, I created a ratio that I used to assign a given sentence quartile label.So for example if a given sentence is a number *13* and the note it belongs to is *50* sentences long, then the ratio is 0,26 assigning to the sentence label *Q2*.

8. **Sentence length** Sentence length may provide some cues on level of detail written into a sentence, especially considering that medical domain sentences can be extreme in their length (in both directions). One can hypothesise that longer sentences may contain more extensive information, such as patient's background and history of functioning, thus making it useful for the classifier to distinguish possible "past" functioning sentences. Similarly, short, to the point, symptom listing sentences would more likely be describing the current health of the patient. Therefore, I decided to use the sentence length feature available already from Kim et al. (2022)'s dataset. The format of this feature is in integer length, such as *6*. I added this feature as it is to the vector representation of the sentence, increasing it by one dimension.

3.5.2 Corpus analysis

To make sure that my chosen features made sense not only based on previous literature, but also grounded in the data at hand, I have looked at some basic distributional statistics of some of them. Specifically, those features are verb, verb tense, temporal expressions, relative sentence position in the note, and sentence length. Looking at distribution statistics of the morphological features is time consuming and difficult to present in a meaningful way for the sake of the goals of this thesis. Here I will mainly discuss the summarised tendencies of the feature label distributions to describe some overall trends within these features in relation to the temporal relation identification task. To see the actual tables with counts, please refer to Appendix B, where I have provided the unsummarised tables used for calculation of relative frequencies in this subsection.

For verb feature, when looking at training set distributions, we can see that only 11,98 % of tokens were labelled as verbs. This percentage is quite stable across the temporal relation labels: 13,34 % for the label *past*, 11,62 % for the label *now*, and 11,51 % for the label *future*. To summarise, I combined detailed verb tags into groupings by their form (see Table 3.6). Across all groups, with the exception of infinitive group, *past* subset had the larger occurrence of verbs relative to other tokens. For infinitive group, interestingly, it was *future* subset. I think that may show usefulness of infinitive verbs as indicators of *future* label in text classification. Similarly, past participle forms being twice as frequent in *past* subset compared to *now* and *future* could be another indicator for differentiation.

For tense feature, we see a slightly clearer distinction in Table 3.7. The relative occurrence of the features is quite low and even throughout the dataset: 8,12 % of tokens in *past* subset, 8,21 % in *now* subset, and 8,46 % in *future* subset. But we

	past %	now %	future %
infinitive	1,89	2,21	3,87
present participle	0,62	0,49	0,55
past participle	4,30	2,66	1,99
finite	6,53	6,27	5,11
total	13,34	11,62	11,51

Table 3.6: Verb form groups relative occurrence in training set.

see four times larger relative occurrence of past tense in the *past* label compared to the other two labels. In similar fashion, infinitive is almost twice as frequent in the *future* label as the other labels. But when it comes to present tense, it seems to be twice as less frequent in the *past* label compared to *now* and *future* labels. So, based on the occurrences in the training data, I think this feature may be quite useful in differentiation of the temporal relation labels.

	past %	now %	future %
infinitive	1,60	1,95	3,36
past	3,68	0,91	0,58
present	2,84	5,35	4,52
total	8,12	8,21	8,46

Table 3.7: Verb tense's relative occurrence in training set.

When it comes to temporal expressions, in Table 3.8 you can see that the NER classifier assigned temporal labels to a very small portion of the tokens. Specifically, only 2,32 % of tokens in *past* subset, 2,25 % in *now* subset, and 2,27 % in *future* subset. When it comes to the labels themselves, both *DATE* and *TIME* are interestingly uniform in their distributions across the temporal relation labels: 1,99 % and 0,33 % respectively in the *past*, 1,92 % and 0,33 % in *now*, and 1,95 % and 0,32 % in *future*. This, however, does not say much about helpfulness of this feature to the classifier. Although, perhaps, the position of temporal expressions within the sentence is indicative of the differences between the subsets. Either way, I decided to still keep this feature because it is one of the highlighted features for this task in literature.

	past %	now %	future %
DATE	1,99	1,92	1,95
TIME	0,33	0,33	0,32
total	2,32	2,25	2,27

Table 3.8: Temporal expressions' relative occurrence in training set.

The next feature is sentence position in the note, as seen in Table 3.9. Interestingly, for the label *past* sentences tend to be positioned in the beginning and the center of the note, with only 6,46 % of the sentences about past functioning located in the last quarter of the medical note. Quite a similar thing can be observed with the *now* subset, although to a smaller scale: the largest portion of the subset is in the second quartile, then almost a quarter of the subset each are in the first quartile, 27,14 %, and in the third quartile, 24,36 %. Again, the smallest portion of the sentences is in fourth quartile - 12,59 % - though still twice as frequent as for the *past* label. For *future* label

the distributions were reversed: the largest portions were in third and fourth quartile, 35,45 % and 34,60 %. Whereas 21,31 % of the *future* label were in the second quartile and only 8,64 % were in the first quartile. Based on these distributions, I believe this sentence position feature is quite useful in distinguishing between the temporal relation labels. At the very least, there is a clear inverse relation in position with *future* label and position with *past* and *now* labels.

	past %	now %	future %
Q1	31,71	27,14	8,64
Q2	31,67	35,92	21,31
Q3	30,16	24,36	35,45
Q4	6,46	12,59	34,60

Table 3.9: Relative sentence positions' distributions by gold labels in training set.

And the last feature I have looked at is sentence length. Unlike previously discussed features, this feature is quantitative, which is why distributions presented in Table 3.10 are more like typically seen statistics. The mean and the median of sentence lengths were around the same for each subset: 122,47 and 112 tokens in *past* subset, 78,97 and 63 tokens in *now* subset, and 67,5 and 56 tokens in *future* subset. It seems that the *past* label sentences tend to be relatively longer than *now* or *future* label sentences. Although, the standard deviation for all of the subsets is quite large: 64,43 for *past* sentences, 61,53 for *now* sentences, and 50,17 for *future* sentences. Especially, seeing that minimum sentence length is extremely short across all subsets - 5 for *past*, 1 for *now* and *future* - and maximum sentence length is extremely long - 831 for *past* and *now* subsets, 432 for *future* subset. So, the differences in the sentence length distributions across the temporal relation subsets are quite volatile.

	past	now	future
mean	122,47	78,97	67,5
median	112	63	56
standard deviation	64,43	61,53	50,17
minimum	5	1	1
maximum	831	831	432

Table 3.10: Sentence length's distributions by gold labels in training set.

Overall, there are no features that seem to clearly differentiate all three subsets of temporal relation labels from each other. But all of the temporal and discourse features seem to at least differentiate one label from the rest: infinitive verb form, infinitive tense, and first and last quartiles for *future* label, past participle form and past tense for *past* label. Therefore, hypothetically there are enough features that would be helpful in distinguishing the minority labels from the label *now*.

To summarise, we have described in detail the specific task at hand, the data and its qualitative characteristics, as well as the two models we are going to train for the task at hand. We also defined the parameters of those models that we will use explicitly, as well as the features chosen for the feature-explicit model, especially the reasoning behind choosing those features and their distributions within the training data.

Chapter 4

Experiments

4.1 Training SVM

Two types of experiments were conducted in this thesis: SVM classifier experiments and MedRoBERTa.nl classifier experiments. As mentioned in section 3.5.1, I have made initial experiments based on the feature representations that I originally intended to use. However, simple count representation of tokens worked poorly with SVM classifier, and its combination with other features too. For example, just count token representation model would work really well on *now* label of temporal relation identification, with f1-score of *.73*, but absolutely terribly on *past* and *future* labels, with f1-scores of *.08* and *.12* accordingly. Addition of the rest of the features did not improve the classifier's performance.

Then, I have tried to train the SVM classifiers on all other features, but without tokens feature. That also did not work well, with *past* and *future* labels classification so poor that their f1-scores both rounded to *0*. Although, that meant that the tokens themselves are still an important feature for the temporal relation identification task. As also discussed in section 3.5.1, I decided to try out TF-IDF weighted representation of tokens. Without any other additional features, the new SVM classifier was able to make much better predictions of the minority labels than before: *past* label's f1-score was *.66*, *future* label's f1-score was *.68*, whereas *now* had f1-score of *.87*. Therefore, I decided to stick to this tokens feature representation and use it for my experimentation.

All of the above experimentations were done with the linear kernel. For interest's sake, I also tried using Gaussian kernel, but the results dropped after addition of the other features besides weighted tokens. Aforementioned experiments are not fully reported on in the next subsection, as they were just try-outs of different parameters and features. Instead, the next subsection reports on the results of 4 experiments done on development and test sets of the data. Specifically, these experiments are the morphological feature experiment, the discourse feature experiment, the temporal feature experiment, and the final, best feature experiment. The morphological feature experiment included the following features: tokens, UPOS, XPOS, and syntactic dependencies. The discourse feature experiment included tokens feature, relative sentence position in the note, and the note length feature. The temporal feature experiment included tokens feature, temporal expressions feature, verb feature, and verb tense feature. And for the final experiment, we basically combined all of the features, because previous three experiments all did relatively well compared to each other.

4.1.1 Results

To create the feature-explicit SVM model I have first conducted three experiments: first experiment being morphological features combination with the tokens, second experiment being the discourse features with the tokens, and the third experiment being the temporal features with the tokens. These experiments were done on the development set of the data. Then, the best groups of features were combined together in the fourth experiments which I conducted both on the development and on the test set.

In Table 4.1, you can see that morphological features in combination with TF-IDF weighted tokens result in similar f1-scores for labels *past* and *future*, *.64* and *.67*, which are decent. Whereas the *now* label has a much higher f1-score of *.86*. Interestingly, precision scores for labels *past* and *future* are higher than their recall scores, with precision scores of *.69* and *.75* compared to recall scores of *.60* and *.60*. *Now* label had the opposite phenomenon: precision score, *.82*, being lower than the recall score, *.90*. Overall, the macro average scores are good, with all of the metrics above *.7*: precision is *.76*, recall is *.70*, and f1-score is *.72*.

	precision	recall	f1-score	support
past	0.688	0.602	0.642	337
now	0.823	0.903	0.861	1561
future	0.754	0.596	0.666	525
accuracy			0.795	2423
macro avg	0.755	0.700	0.723	2423

Table 4.1: Classification report, experiment 1, SVM model on dev set.

In the second experiment, discourse features were combined with TF-IDF weighted tokens and the results of it can be seen in Table 4.2. Label *now* has virtually not changed compared to the first experiment: precision is *.82*, recall is *.92*, and f1-score is *.86*. But *past* and *future* labels' predictions have slightly improved. There are still higher precision scores than recall scores for the minority labels, *.73* and *.59* for *past* label, and *.78* and *.60* for *future* label. And similarly to the first experiment, the opposite is true for the *now* label, where recall is higher than precision. Thus, the macro averages of performance metrics are still pretty good, with precision average of *.78*, recall average of *.70*, and f1-score average of *.73*.

	precision	recall	f1-score	support
past	0.727	0.594	0.654	337
now	0.819	0.915	0.864	1561
future	0.780	0.602	0.680	525
accuracy			0.802	2423
macro avg	0.776	0.703	0.733	2423

Table 4.2: Classification report, experiment 2, SVM model on dev set.

Results of the third experiment combining TF-IDF weighted tokens with temporal features can be seen in Table 4.3. Again, the performance metrics are extremely close to the first two experiments. *Now* label is predicted better than the other two labels, with its precision of *.82*, recall of *.91*, and f1-score of *.87*. The minority labels scored

virtually as well as in other experiments: *past* had precision of *.73*, recall of *.60*, and f1-score of *.66*, whereas *future* had precision of *.76*, recall of *.60*, and f1-score of *.67*. The macro averages then stayed almost the same as in experiment two, with precision of *.77*, recall of *.70*, and f1-score of *.73*.

	precision	recall	f1-score	support
past	0.731	0.596	0.657	337
now	0.823	0.914	0.866	1561
future	0.762	0.602	0.672	525
accuracy			0.802	2423
macro avg	0.772	0.704	0.732	2423

Table 4.3: Classification report, experiment 3, SVM model on dev set.

At the end, it seemed that all three previous experiments performed similar to each other. Thus, since there was no single group of features that worsened the predictions of temporal relation labels by much, I decided to proceed with all the feature groups for the fourth experiment. The fourth experiment then combined TF-IDF weighted tokens with morphological features, temporal features, and discourse features.

The results of the fourth experiment on the development set of data can be seen in Table 4.4. The model in this experiment is still performing comparatively to the other three. The *future* label has stayed consistent in its scores: its precision is *.75*, its recall is *.61*, and its f1-score is *.67*. Similarly, *now* label is doing well too, with precision, recall, and f1-scores of *.83*, *.90*, and *.86* respectively. Interestingly though, the performance on the label *past* slightly dropped, with its precision being *.67*, recall being *.59*, and f1-score being *.63*. The macro averages then also stayed in the similar scores, with precision of *.75*, recall of *.70*, and f1-score of *.72*.

	precision	recall	f1-score	support
past	0.671	0.594	0.630	337
now	0.825	0.899	0.860	1561
future	0.750	0.606	0.670	525
accuracy			0.793	2423
macro avg	0.749	0.699	0.720	2423

Table 4.4: Classification report, experiment 4, SVM model on dev set.

With the experiments proving that all of the features chosen for the SVM model are good for temporal relation identification task, I proceeded to do the final predictions with the SVM model from the fourth experiment on the test set of the data. The performance metrics are reported in Table 4.5. Overall, the SVM model seems to have predicted on test set of the data equivalently to the development set of it.

The macro averages on the test set around the same: precision is *.76*, recall is *.69*, and f1-score is *.71*. Precision score for the *past* label has increased by *.05* while the recall for the *future* label dropped by as much. Otherwise there are no larger variations in scores compared to the development set of the data. Metrics for the *past* label show higher precision score compared to the recall score, with precision being *.73* and recall being *.60*, whereas its f1-score is *.66*. Similarly, predictions on label *future* stayed around the same level as with the development set: precision being *.74*, recall being *.56*, and f1-score being *.64*. The predictions of label *now*, expectedly, were the most

	precision	recall	f1-score	support
past	0.727	0.595	0.655	336
now	0.807	0.905	0.853	1560
future	0.737	0.557	0.635	522
accuracy			0.787	2420
macro avg	0.757	0.686	0.714	2420

Table 4.5: Classification report, experiment 4, SVM model on test set.

accurate, with its precision as *.81*, its recall as *.91*, and its f1-score as *.85*.

4.1.2 Error Analysis

With our final SVM model's predictions on the test set of the data, we can further examine what kind of errors does it make in order to answer the research questions in the section 5. Overall, there were 516 misclassified sentences. Of those, *past* label sentences were misclassified as *now* label in 214 cases and as *future* label in 18 cases. Gold *future* sentences were misclassified as *now* label in 123 instances and they were misclassified as *past* label in 13 instances. For gold *now* label, the SVM model predicted *past* label 91 times and it predicted *future* label 57 times.

To look at the errors from a more global perspective, I have added the normalised confusion matrix for the model's performance in Figure 4.1. I have normalised the counts because my dataset is unbalanced and otherwise the visual representation would not be meaningful to our eyes. The normalisation is done on the gold labels of the test data: you can see that the proportions add up to 1 more or less by the horizontal axis.

The diagonal line from the upper left corner to the lower right corner corresponds to the recall scores on the temporal relation labels: *.60* for *past*, *.91* for *now*, and *.56* for *future*. As expected, the minority labels were most incorrectly predicted as the majority label in the data, *now* label, with *.37* of *past* label and with *.41* of *future* label misclassified for it. Only *.04* part of *past* label was misclassified for *future* and only *.03* part of *future* label subset was misclassified for *past* label. Similarly, for label *now* only *.04* were misclassified as *past* and *.06* misclassified as *future*. That seems to suggest that perhaps the features learnt by the classifier predispose it to predict more sentences under the label *now*: it works really well for that label, but it also makes a lot of mistakes for the other two labels. It could also be that the classifier just despite the features learnt keeps assigning many of the sentences the majority label because the data is so unbalanced. Interestingly though, the labels *past* and *future* rarely get confused for each other. Perhaps, they qualitatively differ from each other more than from the label *now*.

To look at the misclassifications made by the SVM model a bit closer, I have looked at the specific sentences that got misclassified. However, I cannot exactly share the examples on individual scale due to high sensitivity of the data. Therefore, the next best thing I could think of was to look at the corpus statistics of the misclassified sentences based on some of their features, the way that I have looked at the data split in section 3. Some of the tables made for this can be found in Appendix C. There, the simple counts of the feature tags within error samples are stated. These counts were then used by me to create relative frequency tables in this section - to be able to compare the frequencies across the board.

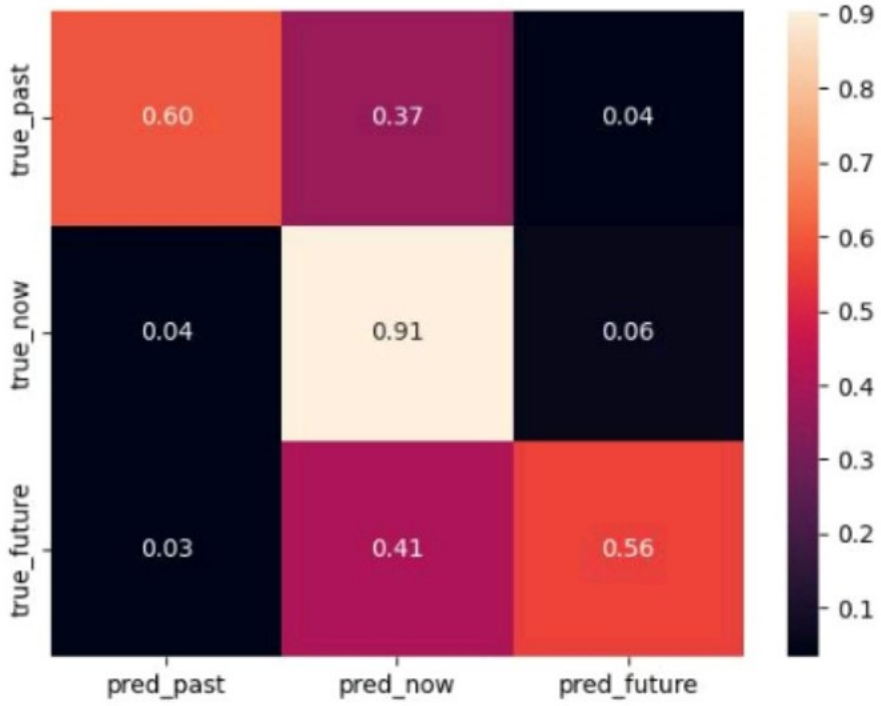

Figure 4.1: Normalised confusion matrix for the final SVM predictions on test set.

When looking at the note lengths of misclassified sentences, in Table 4.6, and comparing them to the distributions of the training set in Table 3.10, we notice some interesting patterns. Misclassified *past* sentences had slightly shorter mean length compared to their training statistics when confused for label *now*, 109,32, and slightly longer mean length when confused for *future* label, 142,78. Although, future sentences were the shorter ones on average in training set, but perhaps the longer and vaguer sentence got, the more likely it could be classified in the hardest to define temporal label - *future*. Misclassified *future* sentences were on average longer when predicted as *past* label, 84,84, and also when predicted as *now* label, 74,12. This may be a marginal difference between the labels or just learnt shortcut in the model's behaviour. Similarly, misclassified *now* sentences were longer than in the training set when confused for *past* label, 107,76, and shorter when confused for *future*, 70,5. Here, logically, numbers make sense, but we should not forget that we are looking at much smaller scale of data and the variations between samples might make it quite hard to compare and draw meaningful tendencies. This also applies to all of the analyses of errors done in this thesis, as it is not easily identifiable what actually causes these errors.

For comparison of verb feature distributions, we can consult back to the Table 3.6 and look here at Table 4.7. Though, it is important to keep in mind that the relative occurrence of the verbs in sentences is quite low to begin with. As with the training set, misclassified *future* sentences had higher rates of infinitive verb forms, which perhaps show that they were not a good differentiator between the labels. Present participle and past participle also do not seem to correlate with the misclassifications made by the SVM model. But interestingly, both misclassified *past* and *now* sentences that were predicted as *future* label had much lower occurrence of finite verb forms, 2,12 % of the misclassified *past* for *future* tokens, and 3,53 % of misclassified *now* for *future* tokens.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
mean	109,32	142,78	84,84	74,12	107,76	70,5
median	97	160	94	71	92	55
standard deviation	63,72	43,17	35,63	49,2	71,34	63,26
minimum	11	36	12	4	9	4
maximum	357	224	124	248	357	229

Note: column names indicate *gold_predicted*.

Table 4.6: Sentence length's distributions in misclassified sentences by the SVM model.

Then, perhaps the finite verb forms produced more confusions for the model's ability to predict future. Also because, misclassified *future* sentences for the *past* label have the highest relative occurrence of the finite verb forms of all errors - 9,73 %.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
none	87,6	93,12	84,96	87,79	88,11	91,01
infinitive	2,25	1,59	4,43	4,12	1,88	3,03
present participle	0,45	0,53	0	0,59	0,49	0,30
past participle	3,64	2,65	0,89	2,14	3,83	2,12
finite	6,06	2,12	9,73	5,37	5,70	3,53

Note: column names indicate *gold_predicted*.

Table 4.7: Relative frequency of verb feature in misclassified sentences by the SVM model in %.

The next feature which I compared occurrences of in the errors comparably to overall data occurrences is the tense feature that can be seen in Table 4.8 down below and its training set distributions in Table 3.7. Again, like with verb feature, infinitive tense was highly occurring in the *future* sentences even when misclassified. But misclassified *now* sentences for *future* label also had an increased occurrence of infinitive, 2,73 %. Also past tense seemed to be a strong indicator for the model to predict *past* label: both gold *future* and gold *now* sentences that were predicted as *past* had a higher occurrence of past tense, 2,66 % and 2,99 %. Although, misclassified *past* for label *now* had that too. For some reason, misclassified *future* for label *past* sentences had the highest occurrence of present tense tokens, 7,08 %, though they also had the highest occurrence of tenses in general. So perhaps, classifier tends to confuse *future* label for *past* label by general larger amount of tenses and verbs in sentences. Present tense's occurrence in predicted *now* labels was also close to its occurrence in gold *now* labels in training set, with 3,85 % of tokens of misclassified *past* label and 4,86 % of misclassified *future* label.

You can see the relative frequency of the temporal expressions in the misclassifications in Table 4.9, whereas the training set distributions are in Table 3.8. There were not many *TIME* labels in the training set to begin with, around 0,3 %, but it seems to also not be frequent in misclassifications. Similarly, label *DATE* was occurring less than 2 % in training set of the data, and it is also around that frequency in the misclassifications. The only strange spikes are in misclassified *past* for future label sentences, 4,23 %, and misclassified *future* for *past* label sentences, 0,89 %. Although, the sample sizes are so small that I think it could be just by chance.

The last feature that I examined in misclassifications is the position of the sentence

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
infinitive	1,96	1,59	4,43	3,68	1,60	2,73
past	2,21	0	2,66	0,52	2,99	0,51
present	3,85	2,12	7,08	4,86	2,71	3,33
none	91,98	96,30	85,84	91,32	92,70	93,74

Note: column names indicate *gold_predicted*.

Table 4.8: Relative frequency of tense feature in misclassified sentences by the SVM model in %.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
DATE	1,68	4,23	0,89	1,25	2,43	2,12
TIME	0,25	0	0	0,51	0,70	0,20
none	98,08	95,77	99,11	98,24	96,87	97,68

Note: column names indicate *gold_predicted*.

Table 4.9: Relative frequency of temporal expressions feature in misclassified sentences by the SVM model in %.

in the note, as can be seen in Table 4.10. Its counterpart, relative sentence position in the training set can be found in Table 3.9. Overall, it seems that the misclassified sentences still follow the trends we found in training set of the data: heavy leaning towards the first few quartiles of the note for the *past* label, heavy leaning towards the last two quartiles for the *future* label. And the label *now* kind of mainly stays within second and third quartiles as you would expect with gold labels. There is only slightly more sentences in the second quartile for the misclassified *future* for label *now* and similarly for the first quartile in misclassified *now* for label *past*. This tells me that there is less correlation between predictions made by the SVM model and the relative position of the sentence in the note. Perhaps, the classifier did not learn some of the tendencies this feature has offered us theoretically.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
Q1	32,71	38,89	0	5,69	24,18	17,54
Q2	37,38	22,22	7,69	22,76	21,14	26,32
Q3	26,17	22,22	61,54	38,21	24,39	38,60
Q4	3,74	16,67	30,77	33,33	10,57	17,54

Note: column names indicate *gold_predicted*.

Table 4.10: Frequency of relative sentence position feature in misclassified sentences by the SVM model in %.

To summarise, it seems that temporal expressions feature and relative sentence position feature were not used effectively in learning of the SVM model. However, the tendencies found among note length feature, verb feature, and tense feature may also be just coincidental, as the error sample sizes are quite small and the unbalanced dataset is leading to misclassifications to heavily lean into the majority class. Thus, we cannot easily explain the errors occurring between labels *past* and *future*, but we can speculate on the errors made between other labels.

4.2 Fine-tuning MedRoBERTa.nl

To create an LLM-based temporal relation identification classifier, we fine-tuned pre-trained MedRoBERTa.nl on our training set of the data. As mentioned in section 3, all the training parameters were kept at default for two reasons: firstly, Kim et al. (2022), previous researchers who fine-tuned this model, have not changed their parameters much, and secondly, the concerns for overfitting on such small dataset. Thus, after fine-tuning for 1 epoch we have kept the model as it is and tested it on the development set of the data. After seeing that it predicted temporal relation labels well, we tested it on the test set of the data and described the results in the following subsection.

4.2.1 Results

Predictions of temporal relation labels made by fine-tuned MedRoBERTa.nl model on the development set of the data can be seen in Table 4.11. Overall, the model did well on the task, as you can see by its macro averages that are all above *.7*: its average precision is *.80*, its average recall is *.77*, and its average f1-score is *.78*. Interestingly, model's predictions of *now* label are comparable to SVM model's, with its precision being *.86*, its recall being *.91*, and its f1-score being *.88*. But model's evaluation metrics all with the exception of one are above *.7*, showing a more consistently good performance across the board. The only exception is the recall of the label *future*, *.66*, which is also the lowest metric for the SVM model too. Nevertheless, fine-tuned MedRoBERTa.nl has marginally better scores than the SVM model on label *past*: its precision is *.75*, its recall is *.74*, and its f1-score is *.74*. Similar picture you can see with label *future*, with its precision of *.78*, its recall of *.66*, and its f1-score of *.72*.

	precision	recall	f1-score	support
past	0.748	0.739	0.743	337
now	0.858	0.907	0.882	1561
future	0.784	0.657	0.715	525
accuracy			0.830	2423
macro avg	0.797	0.768	0.780	2423

Table 4.11: Classification report, fine-tuned MedRoBERTa.nl on development set.

Performance of the fine-tuned MedRoBERTa.nl model on the test set of the data is not very different from the development set, with its scores only slightly dropping in comparison. Again, the *now* label stays consistently the best predicted label: its precision is *.86*, its recall is *.90*, and its f1-score is *.88*. The label *past* still has all its component scores above *.7*: its precision is *.71*, its recall is *.75*, and its f1-score is *.73*. Similarly, predictions of the label *future* are around what they were on the development set, with precision of *.79*, recall of *.64*, and f1-score of *.70*. This meant that the macro averages of the scores also all stayed quite similar to development set results, with precision being *.78*, recall being *.76*, and f1-score being *.77*.

4.2.2 Error Analysis

Similarly to the SVM model, when it comes to predictions on the test set of the fine-tuned MedRoBERTa.nl, we can look closer at the misclassifications that the model makes. Fine-tuned MedRoBERTa.nl has made 431 errors. Of those errors, label *past*

	precision	recall	f1-score	support
past	0.711	0.747	0.729	336
now	0.855	0.900	0.877	1560
future	0.786	0.637	0.704	524
accuracy			0.822	2420
macro avg	0.784	0.762	0.770	2420

Table 4.12: Classification report, fine-tuned MedRoBERTa.nl on test set.

got misclassified the most in absolute counts: 168 sentences were predicted as *now* and 22 sentences were predicted as *future*. For *now* gold label, the model predicted 76 instances as label *past* and 80 instances as label *future*. And the least misclassifications in absolute terms happened for the gold label *future*: 15 sentences were predicted as *past* and 70 sentences were predicted as *now*.

However, I have included a normalised confusion matrix in Figure 4.2 to demonstrate relative occurrence of errors in our unbalanced test set. Again, just like with the confusion matrix in Figure 4.1, the diagonal line from the upper left corner to the lower right corner corresponds with the recall scores for the subsets of the temporal relations labels: *.75* for label *past*, *.90* for label *now*, and *.64* for label *future*. As with the SVM model, misclassifications for label *now* are quite rare, with *.05* of gold labels being classified as *past* and the same proportion as the *future* label. And also fine-tuned MedRoBERTa.nl misclassifies minority labels most frequently for the majority label: *.21* of true *past* label and *.32* of true *future* label. This seems to still be due to the unbalanced distribution of the labels in the dataset. But the label *past* was only confused for the label *future* for *.04* of true *past* labels. The same proportion of true *future* labels was misclassified as *past* label.

It is hard to say what exactly is learnt by LLMs during fine-tuning on a linguistic task and whether much of important linguistic information necessary for the task is captured by the classifier. But nevertheless, to take a closer look at the errors that fine-tuned MedRoBERTa.nl made on the test set, I decided to look at the very same feature distributions that I looked at for the SVM model. The reason why I wanted to do so is that these features, theoretically, would be representative of the type of linguistic information that ideally a language understanding model would learn for the temporal relation identification task. These features and their distributions in the training data were discussed in section 3. The raw counts of the features in the error samples of the fine-tuned MedRoBERTa.nl, just as with the SVM model, are included in the Appendix C. There, just as with the SVM model, the counts were used for construction of the relative frequencies presented in the tables of this section.

First feature I have looked at is the note length feature (its statistics can be seen in Table 4.13). This is because I hypothesised that the longer note may include larger amount of background information on the patient's health, thus separating the label *past* from the other two labels. Although, misclassified *past* sentences were closer in their mean length to their training set average rather than to the other two labels: misclassified for *now* was 113,4 tokens long and misclassified for *future* was 110,66 tokens long. Misclassified *now* for *past* label sentences were though close to the training set *past* label sentences - 105,33 tokens long. Similarly, misclassified *now* for *future* sentences were closer to the training set *future* sentences in mean length - 58,37 tokens long. However, when it comes to misclassified *future* sentences, they were generally

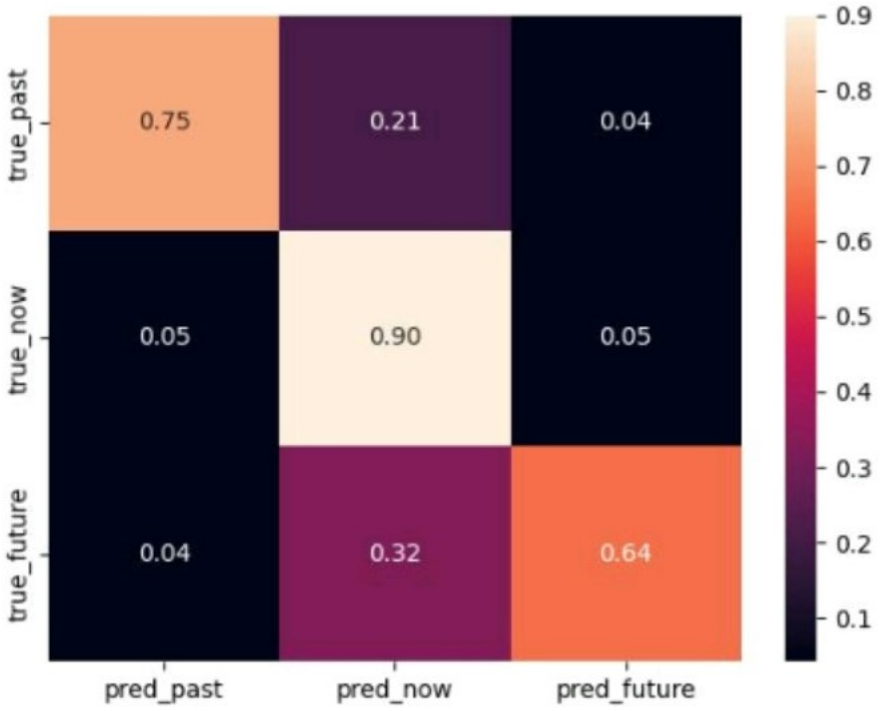

Figure 4.2: Normalised confusion matrix for the fine-tuned MedRoBERTa.nl.

not very far off from the training set *future* sentences in mean length: misclassified for *past* sentences were on average 79,57 tokens long and misclassified for *now* label - 66,31 tokens long. But again, it is important for us to keep in mind that these mean lengths are calculated on quite small samples of errors and the standard deviations, just like in the training set, are quite large - up to 62,27 tokens. Thus, even though misclassifications of *now* gold labels seem to be correlated with the note length, these observations still need to be taken with a grain of salt.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
mean	113,4	110,66	79,57	66,31	105,33	58,37
median	99	103	94	56	88	49
standard deviation	62,27	55,12	29,9	47,52	60,56	51,21
minimum	11	16	17	7	6	3
maximum	357	239	160	197	357	229

Note: column names indicate *gold_predicted*.

Table 4.13: Sentence length's distributions in misclassified sentences by fine-tuned MedRoBERTa.nl.

The next feature is the verb feature, as presented in Table 4.14 for the misclassifications and in Table 3.6 for training dataset. Overall, the ratio of verbs to other tokens is about the same as in training set, keeping to general low frequency. Only misclassified *future* for *past* label sentences have a little less verbs, but it could also be just due to the small sample size. Interestingly, the misclassified *now* for *future* sentences have much higher proportion of infinitive verbs compared to other error samples, 4,22 %, which is closer to the *future* training set proportions. Perhaps, infinitive verbs unexpected

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
none	88,44	89,13	91,77	87,52	88,49	88,06
infinitive	2,12	2,72	1,18	2,67	1,86	4,22
present participle	0,72	0,54	1,18	1,26	0,44	0,29
past participle	3,39	3,26	0	2,38	4,07	2,04
finite	5,33	4,35	5,88	6,17	5,14	5,39

Note: column names indicate *gold_predicted*.

Table 4.14: Relative frequency of verb feature in misclassified sentences by fine-tuned MedRoBERTa.nl in %.

more frequent presence in these *now* sentences is what confused the classifier. Similarly, the predicted *past* label in misclassified *future* and misclassified *now* sentences had the lowest frequencies of infinitive verbs: 1,18 % for *future* and 1,86 % for *now*. Perhaps, the classifier saw infinitive verbs as negatively correlated with the label *past*. Misclassified *past* sentences had lower frequency of past participle verb forms than their training dataset but misclassified *now* for *past* sentences had the highest proportion of them, 4,07 %. The classifier could have learnt to associate past participle verb form with the label *past*. The frequencies of present participle on the other hand have stayed extremely low throughout the misclassifications, just like in training set. Although the slightly higher occurrences in misclassified *future* sentences may be linked with this tag confusing the fine-tuned MedRoBERTa.nl. Similarly, the finite verb form stayed reasonably in the same proportions as in the training data, and the fluctuations in their frequencies between error types do not seem to make much sense.

Tense feature's relative frequencies among the errors can be seen in Table 4.15, whereas its counterpart for training dataset is in Table 3.7. Again, overall the frequency of the tenses across error samples are all quite low, just like in the training set. There is a very clear higher occurrence of past tense in predicted *past* labels: misclassified *future* sentences have 2,35 % and misclassified *now* sentences have 3,37 %. This is in stark contrast with less than 1 % occurrences in the training *future* and *now* subsets, whereas the *past* subset has higher frequencies both in training and even in error rate too. Perhaps the past tense proned the classifier to predict the label *past* more often. Infinitive form has only prominently differed in the error sample of misclassified *now* for *future* sentences, with its high 3,64 %. This is similar to its occurrence in the training *future* label, which makes me think that this form has influenced the classifier to predict the label *future*. Similarly, the present tense in the training data occurred more often in the error samples of misclassified *now* for *future* and misclassified *future* for *now*, 4,8 % and 5,61 %, which is also the case for the training subsets of *now* and *future*. Perhaps, the classifier confuses these two labels due to the high rates of present tense occurrence in both of them.

In Table 4.16 you can see the relative frequency of the temporal expressions feature in the error samples and in Table 3.8 we introduced the feature's frequencies in the training set of the data. Interestingly, the misclassified *future* for *past* sentences had no temporal expressions at all. Similarly, the misclassified *past* for *future* sentences only had *DATE* and no *TIME* expressions. Additionally, misclassified *now* for *future* and misclassified *past* for *future* sentences had the higher than average for training set proportion of *DATE* expressions, 2,33 % and 4,35 %. Maybe, despite not seeing any significant differences in the training set based on the temporal expressions, the

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
infinitive	1,85	2,72	1,18	2,38	1,51	3,64
past	2,08	1,63	2,35	0,56	3,37	0,58
present	3,25	2,72	3,53	5,61	1,77	4,8
none	92,82	92,94	92,94	91,45	93,36	90,98

Note: column names indicate *gold_predicted*.

Table 4.15: Relative frequency of tense feature in misclassified sentences by fine-tuned MedRoBERTa.nl in %.

classifier associates *DATE* expression frequenty with the label *future*. *TIME* has only suspiciously rose in occurrence in the misclassified *future* for *now* sentences, although it is not clear if that would mean this temporal expression is associated with the higher likelihood of label *now* for the fine-tuned MedRoBERTa.nl.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
DATE	1,85	4,35	0	1,26	1,33	2,33
TIME	0,23	0	0	1,26	0,62	0,29
none	97,92	95,65	100	97,48	98,05	97,38

Note: column names indicate *gold_predicted*.

Table 4.16: Relative frequency of temporal expressions feature in misclassified sentences by fine-tuned MedRoBERTa.nl in %.

The last feature I have looked at is the relative sentence position in the note. The misclassified samples' frequencies can be seen in Table 4.17 and the training set frequencies can be seen in Table 3.9. Interestingly, misclassified *future* sentences seemed to still lean more towards the training set *future* distributions, with the lowest frequencies for the first quartile of the note compared to all other error samples. The misclassified *past* for *now* sentences seem to have higher portion of sentences in the second quartile compared to the training set *past* label, whereas misclassified *past* for *future* sentences lean towards the last two quartiles more than the training set *past* label tends to. Similarly, slightly larger proportion of the misclassified *now* for *future* sentences are in the last two quartiles, just like the training set *future* label. So, perhaps, fine-tuned MedRoBERTa.nl has picked up these quartiles as indicators of the possible *future* label. The misclassified *now* for *past* error sample had a higher portion of first quartile sentences which could also suggest the classifier learning that first quartile is highly associated with the label *past*.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
Q1	27,98	13,64	6,67	5,71	25	16,25
Q2	39,88	36,36	26,67	22,86	30,26	18,75
Q3	27,38	22,73	20	40	32,9	40
Q4	4,76	27,27	46,67	31,43	11,84	25

Note: column names indicate *gold_predicted*.

Table 4.17: Frequency of relative sentence position feature in misclassified sentences by fine-tuned MedRoBERTa.nl in %.

In summary, although temporal expressions feature did not seem to have large

differences between temporal relation labels, it seemed to be at least somewhat useful to the classifier. Features note length, verb, tense, and relative sentence position have shown to be used by the classifier in a variety of the situations. However, of course we need to keep in mind that these error samples that we looked at are quite small, making the correlations that I found possibly wishful thinking. The reasons behind the errors are a mystery to us, which makes these observations purely speculative. Further adversarial testing would need to be done on the specific linguistic phenomena that we think the classifier needs for temporal relation identification task in order to say anything conclusive of its comprehension and usage of these features in learning.

Chapter 5

Discussion

In this section I will discuss my main findings in relation to the research questions I have endeavored to answer with this thesis. I will look into the different challenges and limitations of the work I have done and possible solutions for them. I will also look into the area of future research that could be performed in relation to my topic and how can this be useful for the medical center and the ongoing project.

5.1 Answering the Research Questions

To try to answer my research questions stated in the beginning in section 1.2, I will compare the results of my two different systems: trained SVM on my chosen features and fine-tuned MedRoBERTa.nl. After performing three different experiments on the SVM classifier, I have settled on keeping all eight features that I have described in section 3.5.1.

This final SVM model and fine-tuned MedRoBERTa.nl both did quite well on the development set of the data, which is why I also tested both of them on the test set of the data. They both had their macro averages of performance metrics up near *.7* mark. However, the recall of my SVM model was lagging behind slightly with its *.69* score. Overall, the SVM model only had its precision scores for component labels above *.7* and recall and f1-score of the majority class, *now*, above that number. Fine-tuned MedRoBERTa.nl model also had the highest scores for the data's majority class, *now*, which were all above *.8*. But this classifier also had all its component scores above *.7* with the exception of the recall of the label *future*. It would make sense to me that both of the classifiers struggled the most with the label *future*, as it is the hardest one to capture in definitive terms and catch all of its occurrences in the data. *Future* can stretch out to infinity in temporal space, can be probabilistic, and can be discussed in only vague terms and points of time.

Although, we have mentioned the imbalance between temporal relation labels in the data, it is important to remind you that it is the main reason why the label *now* is outdoing other labels in the model predictions. And as mentioned in section 3, I have constructed the dataset according to the natural occurrence of these labels in proportion to each other (minus the negative examples). Therefore, I think in deciding what classifier is better to use for temporal relation identification task performed on this data from the AUMC medical notes that will also be further used in the A-PROOF project only within this hospital, it is important to identify better the minority labels, *past* and *future*.

Thus, in answering the main research question, *What is the more suitable approach to creation of relative time identification classifier in the medical domain?*, I think that in this case, considering our results, the fine-tuned MedRoBERTa.nl for this task may be a better choice. This is because, as I have stated, it is better at identification of the labels *past* and *future*, as well as that we know that the data with which this classifier will be used would be similar to the data we have worked with so far. It is still quite a mystery though how or why can it predict temporal relation labels better than the SVM model that had more metadata available to it.

To answer how do the SVM model and fine-tuned MedRoBERTa.nl compare to each other, we already said that one has better identification of the minority labels. But also looking at the error analyses that we performed, it is interesting to see that the SVM model did not seem to use temporal expressions feature and relative sentence position feature as much in the learning. Weirdly enough, it seemed that the fine-tuned MedRoBERTa.nl actually did use some kind of association with temporal expressions in the sentences, as there were differences between the errors seen in the model's predictions. Both models seemed to be having some correlations between the predictions and the verb, tense, and note length features, albeit usually differentiating between two labels or heavily leaning towards a specific label. There were not features that seemed to clearly differentiate between all three labels. And although the fine-tuned MedRoBERTa.nl is a black-box classifier, we could speculate that if it would learn any theoretically useful features for temporal relation identification, they would be similar to what we have operationalised in this thesis for the SVM model. Additionally, I just think it is important to try and compare these models to similar standards at least a little bit, after all we want the final classifier to capture linguistic concepts and phenomena.

To determine the qualitative differences between subsets of the temporal relation labels in the data, I have performed a corpus analysis on the training set of the data. As I have stated in section 3.5.2, none of the features worked universally as differentiators of all three temporal relation labels. However, the temporal and the discourse features that I have used seemed to at least differentiate two labels from each other at a time. Verb feature had infinitive form and tense feature had the infinitive tense that seemed to be more associated with the label *future*. Interestingly, they were also prominent in the errors of the classifiers, which might mean they were also used as differentiating features. Relative time position feature had the first quartile inversely associated with and the fourth quartile associated with the label *future*. Interestingly, this did not seem to be properly captured by the SVM model, but the fine-tuned MedRoBERTa.nl seemed to have a loose association with it (though perhaps it is something else, as this feature would not be available to this classifier). Similarly, past participle verb form and past tense were good indicators of the label *past* in the training data, which also happened in the classifiers' errors. Thus, it seems that there were definite differences between subsets *past* and *future* that could point one of them out from the rest of the labels, but there were no differences that could differentiate the subsets with clear three way boundaries.

Answering the last two subquestions is a bit out of the scope of my thesis: there needs to be a separate research done in order to build a classifier for absolute temporal relation identification task and there would need to be more data to have meaningful sample sizes to compare temporal relation labels across each of the nine ICF functioning label subsets. Creating a relative temporal relation identification classifier is then the

first step towards accurate placement of the ICF sentences across the patient recovery timeline to track their health functioning progress. Since the relative temporal labels are anchored to the document creation time, to translate them to the absolute temporal labels we would need to try to identify the dates of the ICF functioning within the subsets of *past* and *future*. That way some of the ambiguity is already gone, although, of course there are always error propagation risks. The other future research topics and points of improvement I discuss in the section following the next one on limitations of current research.

5.2 Limitations

There were several limitations when it comes to the experimental set up of this thesis. They mainly stem from the following issues: preprocessing of the raw data, previously made annotations, data sensitivity, and time constraints. Of course, as in any Machine Learning pipeline, there are also limitations stemming from the error propagation of used tools. For example, the data processing package used in preprocessing stage, pandas, has a chained assignment ambiguity issue, which could lead to some of the sentences of the dataset getting lost without me noticing. Similarly, Dutch pipeline of spaCy package has varying levels of accuracy for the components used in feature engineering, section 3.5.1: *.96* for UPOS, *.94* for XPOS, *.85* for syntactic dependencies, *.73* (f1-score) for NER [2].

Preprocessing of the raw data was mainly done by Verkijk and Vossen (2021) and by Kim et al. (2022). Verkijk and Vossen (2021) have de-identified the raw medical notes by replacing all the personal names and references by token *PERSOON* or *person*. Kim et al. (2022) have preprocessed the raw individual medical notes for annotation, then compiled them into larger documents, and again processed the annotated notes to automatically split into sentences. These steps just as with using ready-made pipelines, introduced errors into our data and features used in this thesis.

When it comes to annotations made for the current task, the issue has been already discussed by Kim (2021). Due to time constraints, the annotators were asked to just label the past or expected future ICF statuses of patients on the entire sentence. That is in contrast with the mentions of current ICF statuses that in the annotation tool, INCEpTION, could be labelled on individual tokens as well as the level of ICF functioning being also indicated on a token in the same sentence (Kim, 2021). This limited the scope of features that could be used to build SVM, as event-centered features could no longer be included. Similarly, no temporal expressions annotations were made due to time constraints put on this project: this meant that temporal expressions that could be used were not precise enough to not introduce errors as well.

Data used for this thesis spans years 2017, 2018, and 2020. Although, as mentioned in section 3.2.2, sentences mentioning ICF labels are quite infrequent, using more data could have helped with improving the classifiers for the temporal relation identification task. Of course, that is a long resource-consuming process, but it could help with fine-tuning MedRoBERTa.nl for larger number of epochs without the present risk of overfitting on the training data.

And in general, with larger timeframe, there would be more space for experimentation with hyperparameters of both feature-explicit and black-box classifiers. For

[2] `https://spacy.io/models/nl`

example, even though I have tried out using a different kernel for SVM model - Gaussian kernel - there was nearly not enough time to experiment with other kernels, let alone other parameters of the model. Similarly, doing an exhaustive gridsearch for the optimal hyperparameters for MedRoBERTa.nl would take too long to complete for the purposes of this thesis.

But the most limiting part of the process was the necessary but still extremely frustrating constraint of using myDRE secure servers [3]. As discussed in sections 3 and Appendix A, it is of most importance to ensure zero data leaks in such sensitive domain as medical notes. However, the effectiveness of deployment is questionable: simple Python scripts would take much longer time to run, let alone such intensive processes like fine-tuning a Large Language Model on such a server. And due to firewalls, approval of additional software and resources would take a long time too. This limited the scope of resources that could have been used to improve the classifiers, such as HeidelTime (van de Camp and Christiansen, 2013). Additionally, the security server like this is quite costly and the longer it is used, the larger the bills for it grew.

These limitations have constrained the scope and the quality of research that I could provide on the temporal relation identification at this time. However, considering that, I believe that the results of my experiments and my ability to answer the research questions of this thesis are still solid. In the next subsection I will introduce some of the ways these results could be improved in the future and what further steps can be taken with them.

5.3 Future Works

In case you would wish to endeavour into research on the topic of temporal relation identification in the medical notes, especially in terms of ICF functioning statuses of patients, you could use the following recommendations. I think based on the work done for this thesis, there is still room for improvement in terms of building a relative temporal relation identification classifier for AUMC data. Specifically, trying to improve the obvious limitations of current research by investing more time into annotations of ICF statuses on the token level for labels *past* and *future* would be immensely helpful for finer-grain classification and further building absolute temporal relation classifier. In the same way, putting annotation efforts into labelling gold temporal expressions in the data could help for both relative and absolute temporal relation identification tasks.

In general, as mentioned earlier with my research subquestions, to compare if there are also qualitative differences between ICF statuses in temporal subsets would be useful for learning more about the characteristics of patient health discourse in Dutch medical notes. Thus, annotating more data, from the past three years, for example, could help by increasing the amount of positive examples for the machines to learn from. Of course, we know that annotations are costly, but they may be very well worth the effort in building the foundations for the quality research that A-PROOF project is trying to achieve by building automated recovery timelines of patients.

With more time spent on this topic, exploring the effects of training the model with different features from those used in this thesis, as well as trying out other parameters and maybe even model types - perhaps Conditional Random Fields would work better for this task than Support Vector Machines. Similarly, fine-tuning MedRoBERTa.nl

[3] `https://mydre.org/`

with different hyperparameter settings, especially trying to fine-tune it for more epochs with a larger dataset may yield better results. Additionally, looking into adversarial testing of the fine-tuned MedRoBERTa.nl could be a good idea - in order to check whether the model can comprehend the linguistic concepts prerequisite to the temporal relation identification task. From the possible features to try out, if the annotations would be made on the token level, event based features, centered around the ICF functioning statuses, could work better, according to the literature I have read. And if the temporal expressions annotations would be made, they could also be used for predictions of temporal relation labels instead of temporal tags of Named Entity Recogniser that I have used in this thesis. Although, alternatively, this rule based temporal expressions tagger, HeidelTime, developed for the Dutch language by van de Camp and Christiansen (2013), could work well enough instead. Similarly, exploring in more details feature engineering for the verb and tense features could potentially improve the classifier too.

The feature-engineered model could also be improved by changes in feature representations. Although, the features make theoretical sense and their distributions match in the data, we know that machines do not see our reasoning for including them in their learning. Thus, for example, trying to reduce dimensions of the feature representations could help classifier to improve in its performance. One such reduction in dimensionality could be applied to the token representations: we used TF-IDF weighting for the counts of tokens in the data, where term frequency was calculated across the entire training set of the data. An interesting comparison would be if we would calculate the TF-IDF weighting of tokens per subset of temporal relation label: the token would be weighted higher if it occurs more often in the subset *past* for example compared to the other two temporal relation subsets, and if the token belongs to a sentence where the gold label is *past* then. That way unique to the temporal relation subset vocabularies could be made to improve the predictions of temporal relation labels. Similarly, you could try to just put a threshold for TF-IDF weighting to also reduce dimensions.

Besides the improving of relative temporal relation classifier, the next step would be to build absolute temporal relation classifier to add to the A-PROOF project pipeline. Identifying specific dates of ICF functioning statuses of patients can be done in the pipeline after the relative temporal relation classifier. This way, the recovery patterns of the patients can be more accurately created and further automated to help the medical staff of the AUMC.

Chapter 6

Conclusion

In this thesis we have investigated how to construct relative temporal relation identification classifier for medical domain. This research was done as part of the ongoing A-PROOF project at the Amsterdam University Medical Center (The A-PROOF Project, 2023). We have compared two types of Machine Learning approaches used in NLP tasks: the feature-engineering, with the Support Vector Machines model, and the fine-tuning of a medical domain Large Language Model, with fine-tuning MedRoBERTa.nl. Specifically, I have trained the models on the ICF functioning statuses dataset used by previous researchers of the project: medical note sentences taken from EHR system of AUMC and annotated for the ICF statuses identification (Kim et al., 2022). We have created two separate classifiers that overall performed well on the relative temporal relation task. However, the fine-tuned MedRoBERTa.nl did better, with its macro average f1-score of *.77*, which is why we recommend proceeding with using that model as part of the automated recovery pattern reconstruction pipeline that is being created as the end goal of the A-PROOF project.

We do continue to caution you against taking any hasty decision about the generalisability of the current research results to other hospitals or overall medical domain. As each part of this research was carefully constructed to fit the AUMC medical note data, it is very well likely that the classifier has been tuned to this specific type of language - Dutch medical notes of the Amsterdam medical professionals. We have tried to also characterise the data itself, where it comes from and how it was collected, as much as the privacy around it could let us. But also what kind of qualitative linguistic characteristics differentiate the medical note language use when medical professionals describe their patients' health in relation to their disease recovery timeline. We have found, both theoretical and in the training data, some indicators of associations between temporal and discourse linguistic features and the temporal relation labels of ICF statuses.

This research has been successful with consideration for the constraints and limitations we have encountered in its process, but it is only a step within the goals of the A-PROOF project. Hopefully, this research will contribute to further expansion of temporal extraction research in the medical domain, especially the very difficult task of absolute temporal relation identification. This indeed would be an important step in the medical domain NLP, especially in the continued goal of the A-PROOF project: mapping out the recovery trajectories of the patients and determining patterns in their recovery by disease, symptoms, ICF statuses and level.

Appendix A

Data statement

1. **Header**

   Title Clinical notes from Electronic Health Record system of Amsterdam University Medical Center for recognition of function recovery patterns of patients

   Curators A-PROOF core team

   Dataset version Unknown

   Citation Kim et al. (2022)

   GitHub repository `https://github.com/cltl/a-proof-zonmw`

2. **Executive summary**

   This dataset has been created as part of a project to automate functional recovery patterns of patients in the hospitals. The dataset consists of sentences extracted from clinical notes made by medical professionals in Amsterdam University Medical Center that are written mainly in Dutch language. The annotated dataset mainly consists of 15000 sentences annotated on International Classification of Functioning, Disability and Health (ICF) categories and levels at the time of writing the notes, as well as 7500 sentences annotated as discussing past or expected functioning relative to the time of writing.

3. **Curation rationale**

   The larger dataset of clinical notes collected from Electronic Health Record (EHR) system of Amsterdam University Medical Center (UMC) was made in order to create a machine learning classifier that could identify functioning levels of patient's according to International Classification of Functioning, Disability and Health (ICF). This data statement mainly discusses the annotated subset of sentences taken from this constantly updated dataset of clinical notes.

4. **Documentation for source datasets**

   To read about a larger dataset's distributions, refer back to Kim et al. (2022).

5. **Language varieties**

   The language of the data mainly consists of Dutch, with insertion of English and Latin, due to the medical language terminology.

6. **Speaker demographic**

   The dataset does not specify the number of speakers or any specific background of them (such as age, race, or gender). However, we know that the clinical notes come from various medical professionals, such as doctors, nurses, dietitians, general practitioners, etc. Thus, we can presume that they are at the very least: highly educated, have a high level of proficiency in the languages used in the medical notes, and belong to at least middle class.

7. **Annotator demographic**

   There were 6 annotators from the body of medical students involved from the early project development, as well as 2 annotators from the core A-PROOF team that belong to medical professional background. Therefore, all 8 annotators could be characterised as having a various higher education background, as well as good proficiency in the languages of the dataset. The 6 student annotators are most likely in their 20s whereas the age of the other 2 annotators would be unknown.

8. **Speech situation and text characteristics**

   The situation in which the medical notes were made is most likely either during the appointments, in-hospital check ups, in a hurried digitally typed manner. Therefore, the notes contain short simple sentences with medical professional vocabulary targeting other professionals that would check a patient's medical history retrospectively.

9. **Preprocessing and data formatting**

   To select annotation data from the larger database, first, Kim et al. (2022) applied keyword filters targeting words that the team thought would occur frequently with the mentions of specific ICF functioning categories. These keywords as well as the preprocessing code can be found on the GitHub repository linked earlier. Besides the keyword filtering, they also tried to get balanced subsets of data from different years, as well as the different types of notes in the system. This is also mainly due to the fact that most sentences in the clinical notes are not relevant for annotation, meaning that many of the selected sentences still were 'negative' examples.

   As a result of this filtering, from over 10 million notes from years 2017, 2018, and 2020 the dataset was reduced to just about 6000 notes. These notes consisted of 286000 sentences of which 15000 ended up being annotated with ICF category labels and 7500 ended up labelled as describing background or past functioning and target or expected functioning relative to the time of writing.

   The ICF functioning categories used in annotation were energy level (b1300), attention (b140), emotional functioning (b152), respiration (b440), exercise tolerance (b455), weight maintenance (b530), walking (d450), eating (d550), work and employment (d840-d859). The levels of functioning were labelled on scale 0-4 for most categories with the exception of walking and exercise tolerance that were labelled on scale 0-5. Annotations were made using INCEpTION software to ensure privacy of information (Boullosa et al., 2018).

10. **Capture quality**

    The dataset contains all the annotated notes relevant to ICF categories that the curators and annotators were able to capture. However, the dataset does not include notes discussing functioning of patients under 12 years old, but otherwise due to privacy issues no further demographic information on the patients is available.

    The dataset is not anonymised as it is stored on a secure server and processed for anonymisation further at the model construction stage. The dataset can only be used as part of A-PROOF projects as it is not publicly available.

11. **Limitations**

    It is likely that filtering by keywords did not capture all the possible notes related to ICF categories. Furthermore, for recovery pattern reconstruction (the main objective of the project) the dataset misses the entirety of the year 2019. Additionally, a consistent timeline of notes is only available for 40 patients, the rest of the dataset does not contain notes from the same patients for a long enough period of time.

    The annotations made for some of the ICF categories dominate the labelled dataset, whereas some other categories are barely represented. This also has to do with the larger categories (respiration) being related to symptoms of COVID-19 patients - one of the biggest cohorts in the data, and the smaller categories (work and employment) possibly represented mainly in notes of the specialists (e.g. psychologists) that were not represented well in the larger dataset.The annotations made for past and expected functioning were not annotated further on the specific ICF categories and levels.

12. **Metadata**

    The annotation guidelines can be found on the GitHub repository. The annotation process is further described in Kim et al. (2022).

13. **Disclosure and ethical review**

    This dataset creation as the extension of the A-PROOF project was partially funded by the NWO Spinoza Project, the Corona Research Fund, and the NWO-ZonMW project “Effectiveness of allied healthcare in patients recovering from COVID-19”. The creation and annotation of this dataset was approved by the medical research ethical committee of Amsterdam UMC.

Appendix B

Corpus analysis tables

tag	past	now	future
-	45.257	138.379	27.822
WW|inf|nom|zonder|zonder-n	149	404	160
WW|inf|vrij|zonder	836	3.052	1.057
WW|od|nom|met-e|mv-n	1	2	0
WW|od|nom|met-e|zonder-n	4	21	7
WW|od|prenom|met-e	204	420	121
WW|od|prenom|zonder	45	76	9
WW|od|vrij|zonder	67	241	35
WW|pv|conj|ev	3	13	2
WW|pv|tgw|ev	482	3.142	539
WW|pv|tgw|met-t	787	4.537	454
WW|pv|tgw|mv	216	693	428
WW|pv|verl|ev	1.765	1.291	146
WW|pv|verl|mv	159	136	36
WW|vd|nom|met-e|mv-n	1	5	4
WW|vd|nom|met-e|zonder-n	2	2	0
WW|vd|prenom|met-e	100	463	114
WW|vd|prenom|zonder	131	377	68
WW|vd|vrij|zonder	2.012	3.314	440

Table B.1: Verb feature's tag distributions by gold labels in training set.

tag	past	now	future
-	5.712	17.156	3.463
WW|inf|nom|zonder|zonder-n	20	62	25
WW|inf|vrij|zonder	109	400	157
WW|od|nom|met-e|zonder-n	0	1	0
WW|od|prenom|met-e	33	61	17
WW|od|prenom|zonder	5	15	3
WW|od|vrij|zonder	12	31	7
WW|pv|conj|ev	0	1	0
WW|pv|tgw|ev	48	365	71
WW|pv|tgw|met-t	91	568	63
WW|pv|tgw|mv	29	79	46
WW|pv|verl|ev	260	177	19
WW|pv|verl|mv	22	14	2
WW|vd|nom|met-e|mv-n	1	2	1
WW|vd|prenom|met-e	9	48	11
WW|vd|prenom|zonder	21	51	8
WW|vd|vrij|zonder	216	437	53

Table B.2: Verb feature's tag distributions by gold labels in development set.

tag	past	now	future
-	47.976	143.717	28.782
infinitive	836	3.052	1.057
past	1.924	1.427	182
present	1.485	8.372	1.421

Table B.3: Verb tense's distributions by gold labels in training set.

	past	now	future
DATE	1.041	2.998	613
TIME	173	511	102
None	51.007	153.059	30.727

Table B.4: Temporal expression distributions by gold labels in training set.

	past	now	future
Q1	1.331	3.388	234
Q2	1.329	4.484	577
Q3	1.266	3.041	960
Q4	271	1.572	937

Table B.5: Relative sentence positions' distributions by gold labels in training set.

	past	now	future
mean	122,47	78,97	67,5
median	112	63	56
standard deviation	64,43	61,53	50,17
minimum	5	1	1
maximum	831	831	432

Table B.6: Sentence length's distributions by gold labels in training set.

Appendix C

Error analysis tables

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
none	2141	176	96	1193	1267	901
infinitive	55	3	5	56	27	30
present participle	11	1	0	8	7	3
past participle	89	5	1	29	55	21
finite	148	4	11	73	82	35

Note: column names indicate *gold_predicted*.

Table C.1: Frequency of verb feature in misclassified sentences by the SVM model.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
infinitive	48	3	5	50	23	27
past	54	0	3	7	43	5
present	94	4	8	66	39	33
none	2248	182	97	1241	1333	928

Note: column names indicate *gold_predicted*.

Table C.2: Frequency of tense feature in misclassified sentences by the SVM model in %.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
DATE	41	8	1	17	35	21
TIME	6	0	0	7	10	2
none	2397	181	112	1340	1393	970

Note: column names indicate *gold_predicted*.

Table C.3: Frequency of temporal expressions feature in misclassified sentences by the SVM model.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
Q1	70	7	0	7	22	10
Q2	80	4	1	28	26	15
Q3	56	4	8	47	30	22
Q4	8	3	4	41	13	10

Note: column names indicate *gold_predicted*.

Table C.4: Frequency of relative sentence position feature in misclassified sentences by the SVM model.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
none	1958	164	78	624	999	1210
infinitive	47	5	1	19	21	58
present participle	16	1	1	9	5	4
past participle	75	6	0	17	46	28
finite	118	8	5	44	58	74

Note: column names indicate *gold_predicted*.

Table C.5: Frequency of verb feature in misclassified sentences by fine-tuned MedRoBERTa.nl.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
infinitive	41	5	1	17	17	50
past	46	3	2	4	38	8
present	72	5	3	40	20	66
none	2055	171	79	652	1054	1250

Note: column names indicate *gold_predicted*.

Table C.6: Frequency of tense feature in misclassified sentences by fine-tuned MedRoBERTa.nl.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
DATE	41	8	0	9	15	32
TIME	5	0	0	9	7	4
none	2168	176	85	695	1107	1338

Note: column names indicate *gold_predicted*.

Table C.7: Frequency of temporal expressions feature in misclassified sentences by fine-tuned MedRoBERTa.nl.

	pas_now	pas_fut	fut_pas	fut_now	now_pas	now_fut
Q1	47	3	1	4	19	13
Q2	67	8	4	16	23	15
Q3	46	5	3	28	25	32
Q4	8	6	7	22	9	20

Note: column names indicate *gold_predicted*.

Table C.8: Frequency of relative sentence position feature in misclassified sentences by fine-tuned MedRoBERTa.nl.

Bibliography

J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

Y. Belinkov and J. Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, 2019.

E. M. Bender and B. Friedman. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604, 2018.

B. Boullosa, R. E. de Castilho, N. K. Laskari, J.-C. Klie, and I. Gurevych. Integrating knowledge-supported search into the inception annotation platform. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 127–132, 2018.

J. Devlin, M.-W. Chang, K. Lee, and K. N. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 2018. URL `https://arxiv.org/abs/1810.04805`.

Y. B. Gumiel, L. E. Silva e Oliveira, V. Claveau, N. Grabar, E. C. Paraiso, C. Moro, and D. R. Carvalho. Temporal relation extraction in clinical texts: a systematic review. *ACM Computing Surveys (CSUR)*, 54(7):1–36, 2021.

M. Honnibal, I. Montani, S. van Landeghem, and A. Boyd. spaCy: Industrial-strength Natural Language Processing in Python, May 2023. URL `https://doi.org/10.5281/zenodo.7970450`.

J. Kim. Automated Assignment of ICF Functioning Levels to Clinical Notes in Dutch, 2021. `https://github.com/cltl/a-proof-zonmw/blob/main/doc/A_PROOF_final_report_2021-12-08.pdf`.

J. Kim, S. Verkijk, E. Geleijn, M. van der Leeden, C. Meskers, C. Meskers, S. van der Veen, P. Vossen, and G. Widdershoven. Modeling Dutch medical texts for detecting functional categories and levels of COVID-19 patients. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4577–4585, Marseille, France, June 2022. European Language Resources Association. URL `https://aclanthology.org/2022.lrec-1.488`.

K. Kreimeyer, M. Foster, A. Pandey, N. Arya, G. Halford, S. F. Jones, R. Forshee, M. Walderhaug, and T. Botsis. Natural language processing systems for capturing and standardizing unstructured clinical information: a systematic review. *Journal of biomedical informatics*, 73:14–29, 2017.

A. Leeuwenberg and M. F. Moens. Temporal information extraction by predicting relative time-lines. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1237–1246, 2018.

A. Leeuwenberg and M.-F. Moens. Towards extracting absolute event timelines from english clinical reports. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2710–2719, 2020.

C. Lin, D. Dligach, T. A. Miller, S. Bethard, and G. K. Savova. Multilayered temporal modeling for the clinical domain. *Journal of the American Medical Informatics Association*, 23(2):387–395, 2016.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993. URL `https://aclanthology.org/J93-2004`.

R. Maritz, D. Aronsky, and B. Prodinger. The international classification of functioning, disability and health (icf) in electronic health records. *Applied clinical informatics*, 8(03):964–980, 2017.

A. Névéol, H. Dalianis, S. Velupillai, G. Savova, and P. Zweigenbaum. Clinical natural language processing in languages other than english: opportunities and challenges. *Journal of biomedical semantics*, 9(1):1–13, 2018.

J. Nivre, M.-C. De Marneffe, F. Ginter, J. Hajič, C. D. Manning, S. Pyysalo, S. Schuster, F. Tyers, and D. Zeman. Universal dependencies v2: An evergrowing multilingual treebank collection. *arXiv preprint arXiv:2004.10643*, 2020.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

S. Petrov, D. Das, and R. McDonald. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2089–2096, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2012/pdf/274_Paper.pdf`.

T. C. Rajapakse. Simple transformers. `https://github.com/ThilinaRajapakse/simpletransformers`, 2019.

K. Roberts, B. Rink, and S. M. Harabagiu. A flexible framework for recognizing events, temporal expressions, and temporal relations in clinical text. *Journal of the American Medical Informatics Association*, 20(5):867–875, 2013.

W. F. Styler IV, S. Bethard, S. Finan, M. Palmer, S. Pradhan, P. C. De Groen, B. Erickson, T. Miller, C. Lin, G. Savova, et al. Temporal annotation in the clinical domain. *Transactions of the association for computational linguistics*, 2:143–154, 2014.

S. Suster, S. Tulkens, and W. Daelemans. A short review of ethical challenges in clinical natural language processing. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 80–87, 2017.

The A-PROOF Project. The A-PROOF Project, 2023. URL `https://cltl.github.io/a-proof-project/`.

N. UzZaman and J. Allen. Trips and trios system for tempeval-2: Extracting temporal information from text. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 276–283, 2010.

N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, and J. Pustejovsky. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, 2013.

M. van de Camp and H. Christiansen. Resolving relative time expressions in dutch text with constraint handling rules. In *Constraint Solving and Language Processing: 7th International Workshop, CSLP 2012, Orléans, France, September 13-14, 2012, Revised Selected Papers 7*, pages 166–177. Springer, 2013.

F. Van Eynde. Part of speech tagging en lemmatisering van het d-coi corpus. *Intermediate, projectinternal version*, 2005.

G. Van Noord, G. Bouma, F. Van Eynde, D. De Kok, J. Van der Linde, I. Schuurman, E. T. K. Sang, and V. Vandeghinste. Large scale syntactic annotation of written dutch: Lassy. *Essential speech and language technology for Dutch: results by the STEVIN programme*, pages 147–164, 2013.

M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, J. Moszkowicz, and J. Pustejovsky. The tempeval challenge: identifying temporal relations in text. *Language Resources and Evaluation*, 43:161–179, 2009.

M. Verhagen, R. Sauri, T. Caselli, and J. Pustejovsky. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62, 2010.

S. Verkijk and P. Vossen. Medroberta. nl: a language model for dutch electronic health records. *Computational Linguistics in the Netherlands Journal*, 11:141–159, 2021.

R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini, M. El-Bachouti, R. Belvin, and A. Houston. Ontonotes release 5.0. *LDC2013T19, Philadelphia, Penn.: Linguistic Data Consortium*, 2013.

World Health Organization. International Classification of Functioning, Disability and Health (ICF), 2023. URL `https://icd.who.int/dev11/l-icf/en`.