Constructing an NLP-Pipeline to Extract Diagnoses and Diagnostic Evolutions from Clinical Notes in Psychiatry

Jelle Joost Ykema

7743629

Master Applied Data Science, Utrecht University

July 4, 2024

supervised by:
dr. Rosanne Turner
drs. Magali de Rooy
prof. dr. Floortje Scheepers

Acknowledgements

I would like to express my gratitude to dr. Rosanne Turner and drs. Magali de Rooy for supervising, and guiding this project towards fruition, as well as providing feedback at an extraordinary rate. Every remark has been thoughtful and helpful. I am also grateful to the examiners, prof. dr. Floortje Scheepers in particular, who was constructive in communication concerning the bureaucracy.

Special thanks to dr. Daan Kolkman, for coordinating the thesis project (among others), and for speedy communication when issues arose.

Lastly, I would like to thank my cohort, Emma van Ginkel and Marit Hoek, for sparring and thinking about concepts throughout the project.

Abstract

While research in using natural language processing for mining text and coherent concepts from electronic health records within generic healthcare is steadily increasing, psychiatric and mental health care demands a more sophisticated approach due to its ambiguity in texts and is therefore considered to be more difficult. This difficulty is predominantly experienced when analyzing large quantities of letters. Specialists within the UMC Utrecht psychiatry department recognize the advantage of being able to quantify and conceptualize disorders on a large scale, leading to disorder profiles that can be used for further research.

This paper proposes several techniques to extract psychiatric disorders from outpatient- and discharge letters, built on existing frameworks and transparent model-independent processes. These model-independent processes consist of rule-based segmentation and extraction of annotated disorders and disorder status within these letters. Mapping disorders and disorder status has been tackled with different techniques, going from least complex to more complex.

The least complex method, which consisted of a syntactic pattern-matching algorithm called ContextD, supplemented with several rules to fit more nuance for psychiatric texts, performed best on average. Only on precision, the ensemble method based on a majority voting classifier performed better than this purely rule-based approach. The other models within this ensemble classifier are a two-stage method classifying the disorders on a POS-tagged TF-IDF token window masking the patterns from the ContextD rule set using SVM, and a pre-trained transformer model called MedRobBERTa finetuned for negation classifying the disorders on the token window. The last model performed worst overall, with recall being below 0.5.

Analyzing psychiatric outpatient- and discharge letters in such a way has not been done before within this context, this project acts as the foundation where the platform for specialists helps advance analysis by clinical specialists.

Contents

1 Introduction

Within mental healthcare, up to 70% of data concerning clinical research and practice is considered to be in free-text, instead of a pre-structured format [Velupillai et al. 2018]. Psychiatric practice is about diagnosing disorders following detailed mental profiles resulting from dialogue, observations, or functional tests. Within this practice, there is an overall lack of available biomarkers and "golden standard tests" pointing to distinct diagnoses when compared to other fields of medicine [Pies 2007].

These free-text notes, or *clinical notes*, therefore hold valuable salient information about but not limited to the development of diagnoses, the considerations of other diagnoses, or other complications [Velupillai et al. 2018]. Obtaining insight into the objectivity of these notes by mining the text could highlight patterns and/or considerations of diagnoses, and could prevent further burdens on the already stressed psychiatric system in the Netherlands[1].

Rezaii, Wolff, et al. (2022) put it as: *making the subjective objective*. They claim that while the disorder categories in psychiatry are supposed to be distinct, categorizing and diagnosing these disorders following symptoms (which are often apparent as being objective) present a challenge for specialists. This objectivity is supposed to arise from the way these disorders are classified, namely via the DSM-5. This manual is the principal authority for psychiatric diagnoses within the US, and the main source of categorization within the UMC Utrecht concerning psychiatric disorders. Some specialists argue that this way of categorizing disorders is sub-optimal, and that "diagnostic categories [...] require more of a flexible approach of symptoms and causes" [HGR 2019]. Following this, it stands to reason that classification, within the DSM-5 nomenclature, and diagnoses are not the same thing[2].

Insight into these diagnoses by utilizing NLP makes it possible to evaluate the amount of diagnosed disorders or the amount of mentions of a particular disorder, and whether these aspects changed. Other examples for mining distinct features within these notes can be: better prediction for intervention (as stated above) and discovering missed symptoms and/or new categories of disorders. An example of such an application was done in a study by Rezaii, Walker, et al. (2019). They found missing symptoms in a text when trying to predict psychosis with machine learning. Statistical models of language representations found implied concepts that *referenced* symptoms by modeling the semantic content within sentence vectors. They found that "low semantic density" and an *increase* in "talking about voices" were predictors for enabling psychosis.

Another example of NLP applied to mental health practice is a two-stage system that filters keywords in the first stage and regresses critical cases on the filtered text from mental health crisis chatlines. Provided by Swaminathan et al. (2023) in their original paper, this system detects and notes intervention at a significantly faster rate than before the implementation. It brought down the average response time from 9 hours to 8-13 minutes for specialists to facilitate triage.

These examples highlight how such applications help push clinical practice to make better and more appropriate mental health interventions. Another aspect of these cases is that this em-

[1] https://nos.nl/artikel/2515420-wachttijden-geestelijke-gezondheidszorg-nog-altijd-onverminderd-hoog

[2] Henceforth in the report, diagnosed and classified are interchangeably mentioned since classification is a regular term in machine learning nomenclature which directly refers to 'Diagnosed' or 'Not diagnosed' and does therefore not represent the DSM-5 "classification"

phasis on diagnoses makes it possible to highlight experiences with said diagnoses or disorders, which in turn makes it possible to more accurately describe the conditions present.

1.1 Objective

Within the UMC Utrecht psychiatry department, an increase in questions and diagnostic considerations regarding diagnoses on the Autism spectrum and Bipolar disorder became apparent. It then begged the question to specialists if these diagnoses could be mined for further insight. During orientation, the observation was made that Borderline disorder is mentioned sparsely throughout clinical notes, which is suspected to be due to stigmatization around the disorder, and the pivot was made to mining several other disorders for which the specialists would like more insight. When having constructed this pipeline, the implementation could have the following forms but is not limited to investigating how many second opinions were requested, and/or whether the considerations made for different diagnoses after a second opinion.

To tackle this problem, a trilateral project was set up, where one part would investigate the ground truth within these clinical notes [van Ginkel 2024], making inference possible. The other two parts investigate the models used for making this inference, with a focus on one part being less complex, more rule-based, and classical in its machine learning, and the last part being more complex and utilizing LLMs [Hoek 2024]. This sub-project investigates the most optimal robust and less complex setup.

The goal of this report is to investigate the characteristics of rule-based, machine learning, and deep-learning systems on these notes to create a transparent pipeline for the extraction of coherent diagnoses from clinical notes out of the Department of Psychiatry within the UMC Utrecht. This translates to the following research question:

"How can a pipeline be developed to accurately and robustly extract disorder status from clinical notes within the UMC Utrecht psychiatry department, and what are the comparative characteristics and performance metrics of rule-based versus more learning-based or combined systems in this context?"

Specialists in this department have trouble quantifying the amount of diagnosed, considered, or negated disorders for large quantities of clinical notes. If this were possible, a whole new range of conclusions could be drawn which would not be possible beforehand. This project acts as the basis for these questions, investigating whether this goal is possible, and how to achieve this goal and answer this question.

1.2 Background

Given free-text, diagnoses need to be found, and the status of these diagnoses needs to be classified. Within the UMCU several projects related to NLP have come out, such as cliNLP [Menger et al. 2024], an extension to the spaCy module[3][Honnibal et al. 2020].

van Es et al. (2023) compared the available classifiers within the cliNLP module on regular clinical notes, comparing rule-based negation detection and learning classifiers. After annotating medical properties, they make the divide between: *experiencer, temporality or negation*.

[3]Spacy is an open-source library for text processing capable of performing on an industrial level. It is also the preferred framework for the cliNLP package.

Every medical concept should fall within these categories. However, when applying models, the evaluation was only done on negated or non-negated. In other texts, this concept is also known as *qualifier detection*. The main corpus provided to the models was a mixture of (semi-) structured notes, and complete free text. The performance of the models on this heterogeneity could therefore also be investigated.

For the rule-based system, the ContextD algorithm was used [Afzal et al. 2014]. This algorithm is based on the logic of the earlier mentioned NegEx algorithm, supplemented with triggers for identifying if a clinical condition is present in the historical-, hypothetical- or 3rd party sense. It seems the model struggles with the historical aspect because the authors claim that the experiencer and negated are identified accordingly, but temporality requires further work. The results of this model varied significantly among the different notes, with the more structured notes being identified accurately. The best performing learning classifier is the generic version of the English RoBERTa fine-tuned to a Dutch clinical corpus called MedRobBERTa [Verkijk et al. 2021] This model stores contextual information the best out of all the models compared.

In the essence of capturing the best characteristics of both aspects, a two-staged model was proposed that combines rules and learning. The motivation for this is a statement made in Afzal et al. (2014): "While machine-learning and rule-based systems showed good performance, the systems using combination approaches produced the best results." The combination chosen is based on a relatively old study [Goryachev et al. 2007] that combined tagging text with NegEx terms and phrases and classifying on POS terms. With time, NegEx has been replaced by ContextD (given these circumstances), and better POS taggers have become available. Given these advancements, an increase in efficiency is expected.

1.2.1 Making inference on systems

Premises must be set to be assumed as true to make inference, therefore, the development of supervised systems and the performance metrics of unsupervised systems annotated sets are needed. This process needs to happen efficiently and concisely [Berge et al. 2023]. Clinical specialists' time is considered valuable and not optimizing the efficiency and inter- or intra-annotator agreeability of the labelling is therefore out of the question. Labelling is often done on Electronic Health Records (EHRs). EHRs are handled more efficiently when it comes to the assistance of diagnoses. Le Glaz et al. (2021) state that: "EHRs [...] are convenient data sources because of their heterogeneity: they combine structured, semi-structured, and free data, and they often use a significantly controlled language containing medical terms that allow the extraction of concepts."

In such complex tasks as mining text from clinical notes, evaluation of the model and error analysis can give valuable insight into the performance and the results. A paper by Velupillai et al. (2018) provides a summary of the challenges of defining evaluation metrics for NLP research with a focus on mental health. They state that to enhance the field further: "more emphasis needs to be placed on rigorous evaluation". They also provide the confusion matrix and error analysis as main instruments to evaluate models, and warn about the intrinsic shortcomings of NLP models with respect to true negatives. These appear to be ignored by common evaluation processes. True negatives indicate the elimination of exposure of case status but are very difficult to extract from text. An emphasis on negation of medical concepts can indicate true negatives, and help spur the ratio.

Negation has been a challenge to correctly identify within the field of NLP for some time.

A paper by Chapman et al. (2001) investigates this problem and provides a relatively simple solution. To introduce the concept of negation, they give as to how negation was handled at that time. They state that: "[...] most phrases indicating negation are stop words in information retrieval systems and are not even used for indexing". Most times, clinicians state diseases (or disorders) that can be ruled out. When ignoring this dynamic, one can only figure out the amount of false positives the system will flag. An example of a clinical note given by Chapman and colleagues that displays the simplest form of negation:

"The chest X-ray showed no infiltrates and EKG revealed sinus tachycardia."

In this sentence, "infiltrates" is supposed to be negated (this would be a true negative), and "sinus tachycardia" is not negated (this would be a true positive), or *experienced*. This example seems trivial but also applies to mental health notes, especially when the main syntactic way of writing these notes is by ruling out disorders. To take into account negation, the authors provide a rule-based regular expression method based on (only) 15 sentences. This algorithm is called "NegEx". Its performance varied significantly among the expressions, suggesting ambiguity in the notes. The annotators were allowed to label it as ambiguous, while the model could not. They attest to the poor performance of some expressions to this fact.

1.2.2 General bias

According to the earlier cited review by Malgaroli et al. (2023), a general bias towards the language used in the clinical notes exists. An overwhelming 88% of the analyzed studies were in English. This raises the question as to the robustness of these studies when relating to other languages/locations, considering this form of selection bias can present itself in all sorts of skewed assumptions.

Referring back to Velupillai et al. (2018), they claim in their review that mental health is relatively understudied. This issue extends to prominent (qualifier) algorithms, such as ContextD or MedRobBERTa (the final name of the RobBERTa model discussed in van Es et al. (2023)), where they seem to be predominantly developed using physician data.

2 Data & methods

This chapter concerns the data utilized in this project, which will be analysed and discussed in terms of modifications. Following the data section, the methodology for the extraction of disorder status will be reviewed.

2.1 Data

The data at hand are discharge and outpatient letters from the psychiatry department within the UMC Utrecht. Dates range from 2012 up until 2022 and the text encompasses several aspects, such as: considerations, medications, recommendations, medical history, etc. Annotation was done on the conclusions of the clinical narratives provided by the UMC Utrecht patient dossier since the conclusion often summarizes all arguments for and against diagnoses [4]. This approach also relieves clinicians of having to annotate the whole text. There are no corrupt records, missing values, or duplicates present since the data is procured from a larger data ecosystem that has provided clean data.

2.1.1 Procurement

The requested data was validated by the data ethics committee within the UMC Utrecht since this project is a sub-project of the bigger approved "PsyData" project and belongs to scientific research. Names, places, and institutions were anonymized using the DEDUCE model [V. Menger et al. 2018], and birth dates were rounded down to the first date of the respective birth year. The full dataset consisted of 17 910 clinical letters, 3 590 discharge letters, and 14 320 outpatient letters to be precise. 250 discharge- and 250 outpatient–letters were sampled at random to be annotated.

Via a formal data format, the following information was given in the letter table: "PseudoID", "LetterID", "Letter Type", "Text", and "DATETIME". The birth year and gender can be linked to the patient table using the "PseudoID".

[4] The method of filtering for the conclusion can be found in section 2.2

2.1.2 Descriptive Statistics

The text of the letters can be analyzed using basic descriptive statistics. This summarizes the characteristics and gives insight into the distribution of the texts. The annotation, as discussed in Paragraph 2.2, was only done on the conclusions. Therefore a split can be seen in Table 1, which highlights the data distribution.

Table 1: Dataset and text statistics such as average amount of words, average amount of sentences, and average word length, with the lower and upper borders of a 95% confidence interval in brackets.

Letter type	**# documents**	**# words**	**# sentences**	**avg. word length**
Discharge letters	3 590	1 364 (1 341, 1 387)	99 (98, 101)	5.6 (5.6, 5.8)
Conclusions		202 (197, 207)	12 (12, 12)	6.0 (6.0, 6.0)
Outpatient letters	14 320	1 217 (1 201, 1 232)	88 (87, 89)	5.5 (5.5, 5.5)
Conclusions		176 (173, 178)	10 (10, 10)	5.8 (5.8, 5.8)
Total	17 910			

As shown in Table 2, the annotated sample has a corresponding distribution of metrics.

Table 2: Descriptive statistics of the annotated sample such as average amount of words, average amount of sentences, and average word length, with the lower and upper borders of a 95% confidence interval in brackets.

Letter type	**# documents**	**# words**	**# sentences**	**avg. word length**
Discharge letters	250	1 321 (1 229, 1 412)	97 (91, 104)	5.6 (5.6, 5.6)
Conclusions		210 (194, 226)	12 (11, 13)	5.9 (5.9, 6.0)
Outpatient letters	250	1 382 (1 265, 1 499)	101 (92, 110)	5.5 (5.5, 5.5)
Conclusions		207 (191, 223)	12 (11, 13)	5.8 (5.8, 5.9)
Total	500			

The clinical texts were annotated and checked for adequate annotator agreement by inter-annotating sets from different annotators and subsequently arguing for or against disagreements. The subsequent step of arguing for or against choices creates a consensus set. This

process ensures a valid intra-annotator agreement, by systematically removing subjectivity from the annotations. These sets were subsequently classified on an intra-document level using a rule-based approach called ContextD, the transformer-based negation detector (in terms of qualifying) developed by van Es et al. (2023), a two-stage approach utilizing the rules from ContextD for feature vectors and classifying from Goryachev et al. (2007), and an ensemble method.

2.2 Annotation

These narratives were provided as free text with HTML breaks. The conclusion was filtered as having up to four double breaks after triggering the conclusion. After filtering, these narratives were flagged for having discussed diagnoses concerning the following aspects: "Diagnosed", "Considered", and "N.A.". An example of annotation on a **fictitious** sentence has been given below:

> Conclusie: Betreft een vrouw, eerder gediagnosticeerd met STRT{ADHD}ADHDd en STRT{dissociatieve stoornis}ANDInvt, laatsgenoemde lijkt inmiddels geen sprake van te zijn. Opgenomen ivm STRT{bipolaire-1-stoornis}BIP1d, DD STRT{schizofrenie}SFRo.

Where the clinical terms are tagged to belong within a category, as well as: diagnosed (d), considered (o), or N.A. (nvt). Annotations were done concerning the following disorders and are per the DSM-5 manual. Each of these categories has its respective pseudo terms and annotation labels.

- ▷ Attention-deficit/hyperactivity disorder
- ▷ Autism spectrum disorder
- ▷ Schizofrenia
- ▷ Borderline personality disorder
- ▷ Bipolar disorder type 1
- ▷ Post-traumatic stress disorder
- ▷ Depressive disorder
- ▷ Other

2.3 Pipeline

The term pipeline loosely refers to the concept from the SpaCy package mentioned earlier. This concept makes it possible to construct a pipeline from several language processing components which can be interchanged. CliNLP is built on top of SpaCy, and some features used in this project lend their usefulness from this interchangeability and interpretability. The pipeline is visualized in Figure 1.

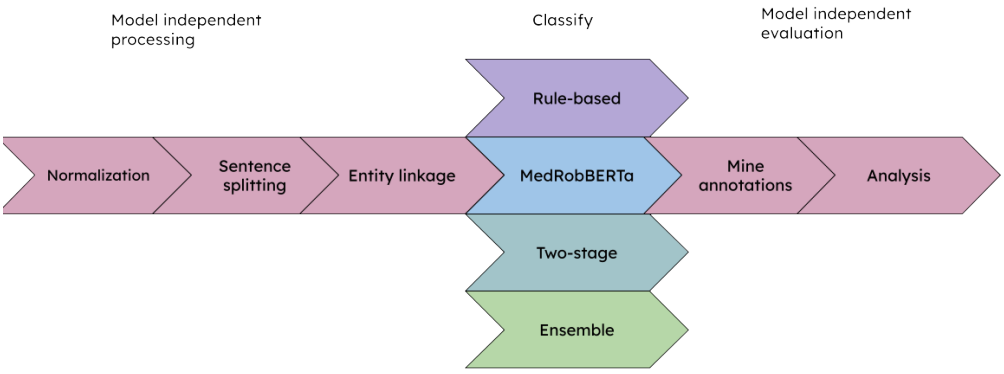

Figure 1: General processing pipeline with model-independent processes and interchangeable qualifiers

At the individual patient level, several diagnoses could be discussed at different points in a narrative. The pipeline is constructed in such a way that it extracts diagnoses, and captures context. These algorithms are being used out of the box and will be tuned after the preliminary error analysis.

The model processes and evaluation are independent of the models so that no bias of these processes can leak to the classifier models, as transparency and differences in model performance are key factors in this process.

2.3.1 Model independent processes

These processes supply a base for the classification models to work on. In Figure 2, an example sentence can be seen.

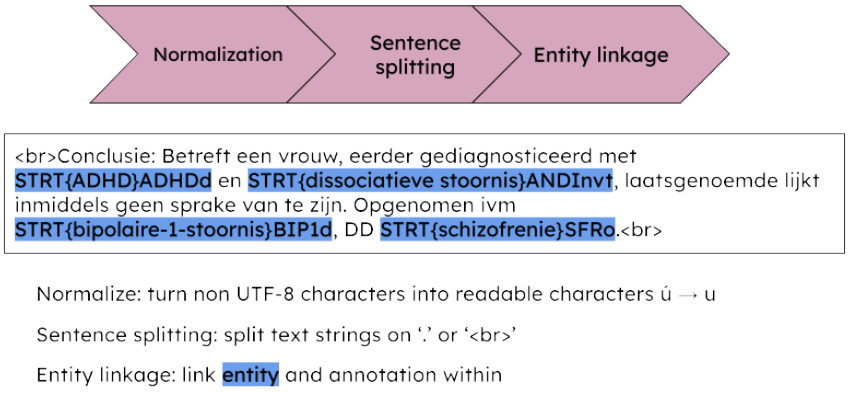

Figure 2: Model independent processes and an example sentence

The normalization of the text needs to be done for the model to make sense of non-UTF-8 characters. Within the clinical notes, these characters appear amply. This feature was used directly out of the cliNLP package. The same goes for the sentence splitting, but this was

supplemented with HTML breaks, to correctly include sentence breaks. For each outpatient- or discharge letter, all sentences were split and were not evaluated if no entity was present.

Entity linkage was done based on a method where first, entities are mined using regular expressions based on the annotations. A rule of thumb is to only mine the entities from the training set, to remain robust. These entities were subsequently linked to a greater subcategory, namely the disorders annotation was done on. The classifiers will see the annotated entities as if they were stripped of the annotation and only the disorder remained.

2.3.2 Rule-based setup

The ContextD algorithm was used for qualifying diagnoses for the rule-based approach. This algorithm is the Dutch version of "Context". It is based on regular expressions scopes out tokens around the tagged concept, and determines whether the concept was negated or not. The most basic version of this principle is called NegEx, an algorithm that seeks out negative or positive phrases around the concept in a 5-token window per sentence, meaning that the text needs to be tokenized and sentenced, this is still a prerequisite for ContextD. An example of the tagging process can be seen in Figure 3.

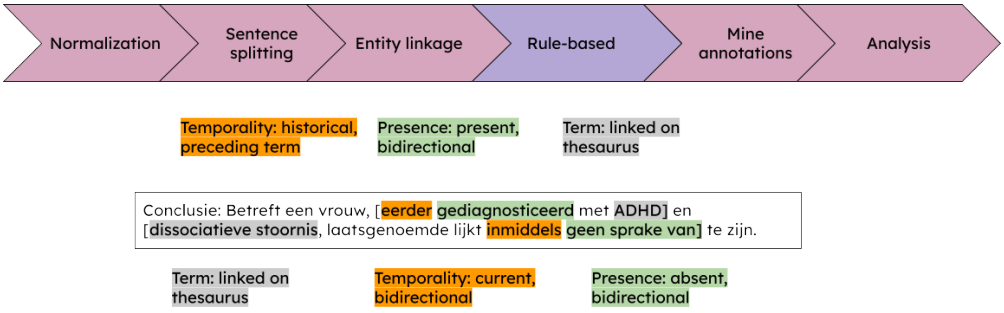

Figure 3: An example of the rule-based approach using the ruleset and algorithm from ContextD

This principle is fairly simple and intuitive, and given the right amount of combinations of rules, can be very effective. NegEx categorizes the concepts into negated or non-negated, while the Context algorithm replaces non-negated with categories like temporality (recent), or experiencer (patient). The algorithm also handles pseudo-negations -terms that imply negation but are not- and terminations triggers -triggers that (should) terminate a context scope. The rules and triggers for this algorithm are extensive, and the main goal is to investigate the performance out of the box, investigate imbalances within these rules and triggers, and adjust accordingly.

The outcome metrics of the ContextD model were classified as having the features as seen in Table 3:

Table 3: Rule-based outcome classifications, corresponding with annotation categories

Annotation category	Presence	Temporality	Experiencer
Diagnosis	Present	Current	Patient
	Present	Historical	Patient
	Present	Future	Patient
Considered	Uncertain	Current	Patient
	Uncertain	Historical	Patient
	Uncertain	Future	Patient
N.A.	Absent	Current	Patient
	Absent	Historical	Patient
	Absent	Future	Patient
	Present | Uncertain | Absent	Current | Historical | Future	Other | Family

Configurable aspects of this model present as custom rules that can be changed within the algorithm, such as supplementary terms or phrases.

2.3.3 Learning model

To capture context by learning, a fine-tuned version of MedRobBerta.nl was provided by van Es et al. (2023). This model was trained on Dutch EHRs and fine-tuned on the "Dutch Clinical Corpus"- or DCC- to detect negation and experiencer. It scopes out tokens around a recognized entity/diagnosis and classifies the context of the term. The outcome metrics pertain to the presence of the diagnosis and the experience, much like in Paragraph 2.3.2. However, it has no temporality module, but this does not directly indicate negation, therefore it can be missed when evaluating. It follows the same categorization as visualized in Table 3, but without the temporality module.

Given the black-box nature of transformers, tuning and adjusting model characteristics can be difficult. However, some aspects can be changed like: the amount of tokens to scope, or re-training the final classification layer on a new labeled set. An example of the tagging process can be seen in Figure 4.

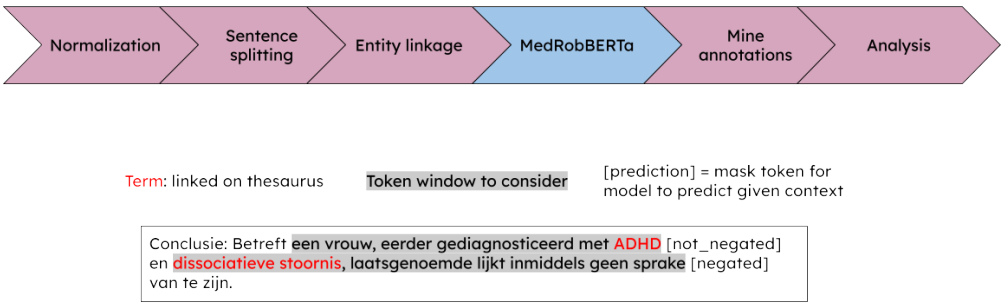

Figure 4: An example of the learning method using the negation detection transformer from van Es et al. (2023)

2.3.4 Two-stage approach

Given a narrative, diagnoses will be linked using a thesaurus. Around this diagnosis, a scope will be determined much like the regular ContextD algorithm, but instead of giving a status to the diagnosis, the whole sentence will be tagged for POS. This transformation will result in POS tags for all terms available where the diagnosis will be masked to be a regular diagnosis. Phrases that co-occur in the ContextD rule set will be flagged as being negated, possible, or experienced in text. Notice that these coincide with the annotation categories. An SVM classifier will be trained on the features that this process creates. This is however not set in stone as the training data set is not that extensive, and comparable performance using Naive Bayes can be expected. Both of these models are lent from the scikit-learn package [Pedregosa et al. 2011]. An example of the tagging process can be seen in Figure 5.

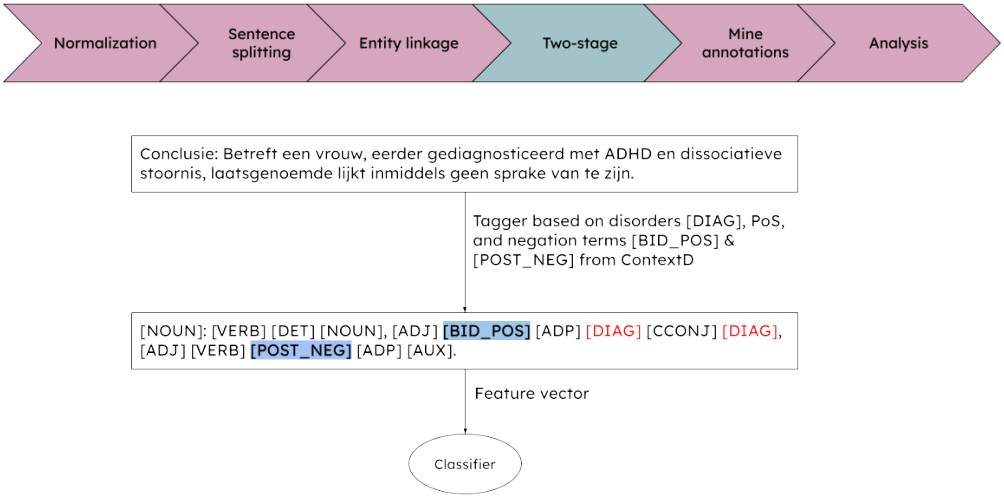

Figure 5: An example of the two-stage method utilizing the trigger phrases from ContextD and a classifier on the feature vectors

The feature vector in this process will be in TF-IDF format, giving more weight to the disorders mentioned and ContextD rules (both of which often appear sparsely), over-generalized POS masks.

2.3.5 Ensemble method

Ensemble learning is a way to stack models and complement or balance out imbalances in prediction. A majority vote classifier was used to stack these models. For evaluating the performance of the classifiers, the annotations need to be mined from the labeled notes, and only then can analysis be done. The mining of these annotations can be seen in an example shown in Figure 6.

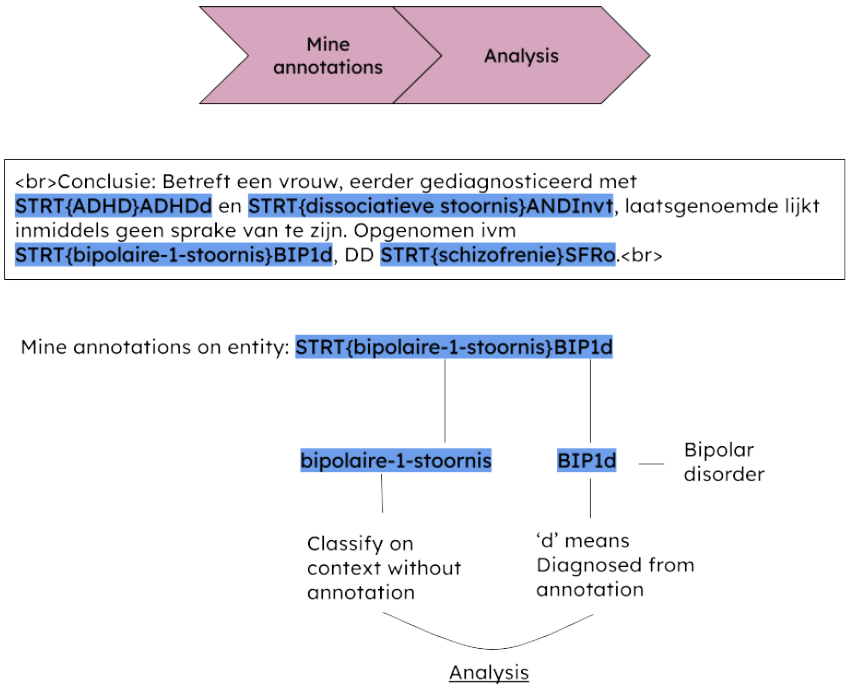

Figure 6: An example of mining the annotation for analysis

The entities contain all the information needed to compare when classifying disorders. Within these entities, the annotations are present for which the classifier trains or gets analyzed. The actual and predicted classifications are put into a confusion matrix. This not only gives insight into the performance but makes stratification on different subsections possible, as well as stratification on disorder.

2.4 Validating models

Because these models are tested out-of-the-box, and a certain bias exists in some models (as discussed in section 1.2.2), they will need to be tuned to more appropriately fit this type of free text. Each model has its tune-able parameters and was briefly discussed in their respective sections. However, they will be summarized here for convenience. The rule-based system can be tuned by expanding or reducing its scope and supplementing rules. The learning-based model can also be tuned by expanding or reducing its scope and/or adjusting classification probability thresholds. And finally, the two-stage model can once again be tuned by expanding or reducing the scope, but can also be tuned by searching over a field of hyperparameter values respective to the classifier.

2.4.1 Structure

A test and train set split was made based on the distribution of annotated entities, such that these splits have the same distribution within. The purpose of this approach was to prevent selection bias in the test and training set concerning the annotation. Benchmarking the out-of-the-box models for performance was done on this split, to compare with tuned versions after validating. For the train set, 5 random subsets are split for the models to train and validate

on. This means that the training set will be randomly split into 5, and 4 of the 5 subsets will be tested against each remaining set, iteratively. This process is called cross-validation. A visualization of the process can be seen in Figure 7

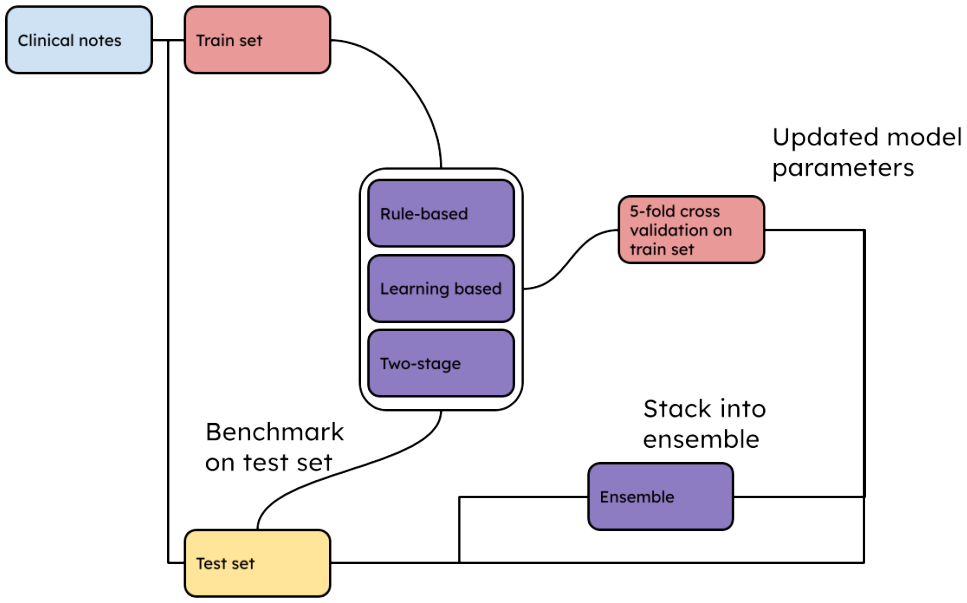

Figure 7: The information flows from the clinical notes through the benchmark and validation.

Parallel to the data flow for validating through cross-validation, the process of iterative vetting on the annotated set can also increase performance. If done correctly, it can align annotated classifications and help achieve better performance overall.

2.4.2 Validation

This section analyses the made errors and highlights the tuned features of the models and the impact these made. A general overview of the classifications as a confusion matrix of all models following 5-fold cross-validation on the annotated set and the consensus set is given in Table 4. Between these two sets, the amount of annotations can differ, meaning that the total sum of entities can be higher or lower, and therefore the sum of the confusion matrix does not necessarily correspond per model.

Table 4: Model benchmarks out of the box for the annotated set and consensus set

	Predicted					
Actual	Annotated set			Consensus set		
	Diagnosed	Considered	NA	Diagnosed	Considered	NA
Rule-based method						
Diagnosed	140	4	7	151	4	5
Considered	20	1	19	41	21	1
NA	20	1	19	25	0	21
Learning model						
Diagnosed	148	0	3	159	0	1
Considered	56	1	1	62	1	0
NA	22	11	7	25	13	8
Two-stage method NB						
Diagnosed	136	3	0	155	3	2
Considered	49	8	0	48	14	1
NA	33	2	5	31	2	13
Two-stage method SVM						
Diagnosed	129	10	0	146	12	2
Considered	28	28	1	30	32	1
NA	26	2	12	27	3	16

From this table, benchmark results show all models classified with a noticeable bias toward classifying a diagnosis. The two-stage models using Naive Bayes and the learning model suffer from this bias the most. Their performance across both sets is very much skewed towards predicting 'Diagnosis'. This seems to be the case for the rule-based model as well on the annotated set without interannotations. However, after inter-annotating and achieving a consensus on this set, ambiguity was most likely worked out and this model went on to perform in a more heterogeneous manner. Across the board, all models still predict 'Diagnosis' too often, seeing 'Considered' as a sort of barrier that acts between 'Diagnosis' and 'NA', resulting in very few (correct) 'NA' classifications.

For tuning the models, metrics such as accuracy, precision, recall, and the F1 score were calculated during validation. These metrics, apart from the recall [5], were calculated in a weighted manner, so that classes that appear scarcely, don't influence the average that much. The recall

[5]Weighted average recall is the same as accuracy

was left unweighted in the averaging to show the spread of the accurate precision over the classifications. For example, as shown in Table 4, the recall across the 'Considered' and 'NA' categories for the rule-based and learning models is significantly worse than for the 'Diagnosed' section. With weighted averaging, this outcome gets muted by the class imbalance. The default configurations in these tables are the same configurations used for prediction as depicted in the right column in Table 4.

In Figure 8 the results can be seen for tuning the rule-based model. Adjusting the scope to take in more tokens and supplementing some basic rules spurred the performance. These rules were mined from looking at misclassifications.

Metric	Max scope of 5, no supplementary rules (default)	Max scope of 7, no supplementary rules	Max scope of 20, no supplementary rules	Max scope of 7, with supplementary rules	Max scope of 15, with supplementary rules
Weighted Accuracy	0.711090	0.722118	0.711399	0.739857	0.736768
Weighted Precision	0.719895	0.723078	0.705397	0.738295	0.733372
Unweighted Recall	0.553830	0.588133	0.595069	0.624237	0.637692
Weighted F1 Score	0.673799	0.695197	0.692581	0.726793	0.729219

Figure 8: Rule based method tuning outcomes, the tuned features here are the rules and the scope around the entity

Misclassifications can mainly be attributed to the models not capturing the essence of the context, addressing them can help make the model more accurate. When looking at these misclassifications for the rule-based model, simple errors were being made where, the rule set from the ContextD algorithm in essence matched the words in the text, but didn't flag them due to it being a synonym. Some examples of an added rule where the default rule captured the essence can be seen in Table 5.

Table 5: Examples of default rules, and supplementary rules in the ContextD rule set

Default rule	Added rule
geen aanwijzingen voor	*onvoldoende aanwijzingen voor*
wordt gedacht aan	*kan worden gedacht aan*
kenmerken van	*trekken van*

In Figure 9 the results can be seen from tuning the learning model. The performance was not susceptible to adjusting the token scope but did react to adjusting the probability thresholds. In specific, adjusting the probability of classifying 'Diagnosed' to a higher probability, was meant to negate the bias that exists for this classification. The same goes for the 'NA' class, where

adjusting this threshold to a higher probability made some impact. Overall, the performance is severely sub-optimal, leaving a lot to discuss.

Metric	NT token window of 32 (0.1 < < 0.9), ET token window of 64 (< 0.5 <) (Default)	NT token window of 10 (0.1 < < 0.9), ET token window of 20 (< 0.5 <)	NT token window of 32 (0.5 < < 0.99), ET token window of 64 (< 0.5 <)	NT token window of 32 (0.55 < < 0.995), ET token window of 64 (< 0.5 <)	NT token window of 32 (0.6 < < 0.9), ET token window of 64 (< 0.5 <)
Weighted Accuracy	0.639116	0.638594	0.592931	0.506882	0.662815
Weighted Precision	0.540205	0.541278	0.578992	0.593984	0.589584
Unweighted Recall	0.406382	0.406658	0.434179	0.430750	0.462444
Weighted F1 Score	0.541416	0.542383	0.562134	0.529999	0.578556

Figure 9: Learning model tuning with transformer properties

In Figures 10 & 11 the tuning of the two-stage models can be seen. These two models fit the data more nudged towards classifying considerations, which is healthy for the heterogeneity between the models. The SVM model responded better to tuning, and with some tweaks, the performance went up significantly. However, the Naive Bayes model was not so susceptible to tuning and performed poorly overall. If these models performed the same, the argument could be made to choose the NB model since this model is less complex compared to the SVM model, but the performance difference is too large.

Metric	5 token window with no smoothing	7 token window with no smoothing	7 token window with alpha = 0.5
Weighted Accuracy	0.656005	0.657320	0.665427
Weighted Precision	0.653661	0.672266	0.665427
Unweighted Recall	0.439162	0.431567	0.444826
Weighted F1 Score	0.585884	0.579586	0.590857

Figure 10: Two stage model using Naive Bayes tuning with several parameters

Metric	5 token window with linear kernel	5 token window with radial base kernel (default)	5 token window with polynomial kernel of degree 5	5 token window with polynomial kernel of degree 1	20 token window with polynomial kernel of degree 1	20 token window with radial base kernel	15 token window with radial base kernel	11 token window with radial base kernel
Weighted Accuracy	0.688629	0.691891	0.668406	0.692542	0.704292	0.716700	0.723231	0.729088
Weighted Precision	0.674057	0.686021	0.645316	0.687939	0.698835	0.713653	0.720094	0.725423
Unweighted Recall	0.527736	0.539540	0.531384	0.540465	0.567184	0.584366	0.596265	0.609873
Weighted F1 Score	0.657498	0.664043	0.649116	0.664847	0.680809	0.694662	0.702546	0.712393

Figure 11: Two stage model using SVM tuning with several parameters

2.4.3 Final models

The final models used for testing and their configurations can be seen in Table 6.

Table 6: Final models with respective configurations

Model	Configuration
Rule-based model	Max scope of 15, ContextD algorithm with complementary rules
Learning model	Negation transformer token window of 32, probability threshold set to $NA < 0.6 < Considered < 0.9 < Diagnosed$ and experience transformer token window of 64, probability threshold set to $Patient < 0.5 < Family$
Two-stage model	SVM classifier with a token window of 11, and a radial base kernel
Ensemble model	Majority vote classifier using the: the two-stage, rule-based, and learning models mentioned in this section, with corresponding parameters

3 Results

In this chapter, the results will be presented of the approaches taken to tackle the problem and reach the objective of constructing the pipeline. A glance will be taken at the performance as a whole.A final error analysis highlighting errors that persist even after iteration will conclude these sections and therefore the results chapter.

3.1 Overview

This paragraph provides an overview of the final models and their respective performance on the test set and a global overview of misclassifications per model. Table 7 provides a summary of the results per model:

Table 7: Performance of the models on the test set with the highest value per category in bold

Model	Accuracy	Weighted average precision	Unweighted average recall	Weighted average F1
Rule-based model	**0.77324**	0.76995	**0.71723**	**0.76851**
Learning model	0.64312	0.56925	0.43074	0.54648
Two-stage	0.73234	0.72641	0.60093	0.71099
Ensemble	0.75939	**0.78088**	0.63821	0.73797

The performance metrics are weighted and averaged where possible, but the recall is left unweighted to highlight an imbalance in misclassifications. Overall, the rule-based method seems to outperform every other method, with the ensemble method having the highest precision and overall second-best metrics. The learning model performed the worst out of all the models, even falling below the 0.5 threshold for the unweighted recall. Figure 12 shows the number of errors made per model and gives insight into how this unweighted recall is distributed among other models.

Figure 12: Total amount of errors or misclassifications made per model

To see how these errors are situated per classification, an overview of the confusion matrices per model was made. This can be seen in Figure 13.

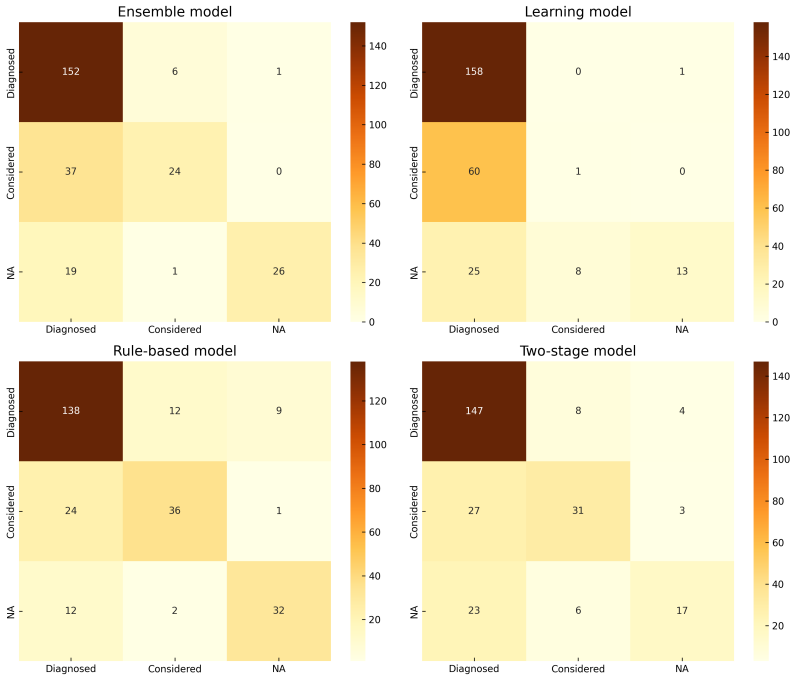

Figure 13: Confusion matrix per model with the x-axis depict predicted -, and the y-axis actual -classes, the main diagonal represent correct classifications

When viewing these matrices, it becomes clear why, for example, the learning model has such bad performance. There are almost no predicted classes that are not 'Diagnosis'. For the rest of the models, the bias that existed earlier, mentioned in paragraph 2.4.2, seems to be less severe but still exists. When stratified on the disorder, as seen in Figure 14, no apparent discrepancies between models can be seen, except for the learning model having a lot of errors.

This figure also gives insight into the distribution of disorders within the letters.

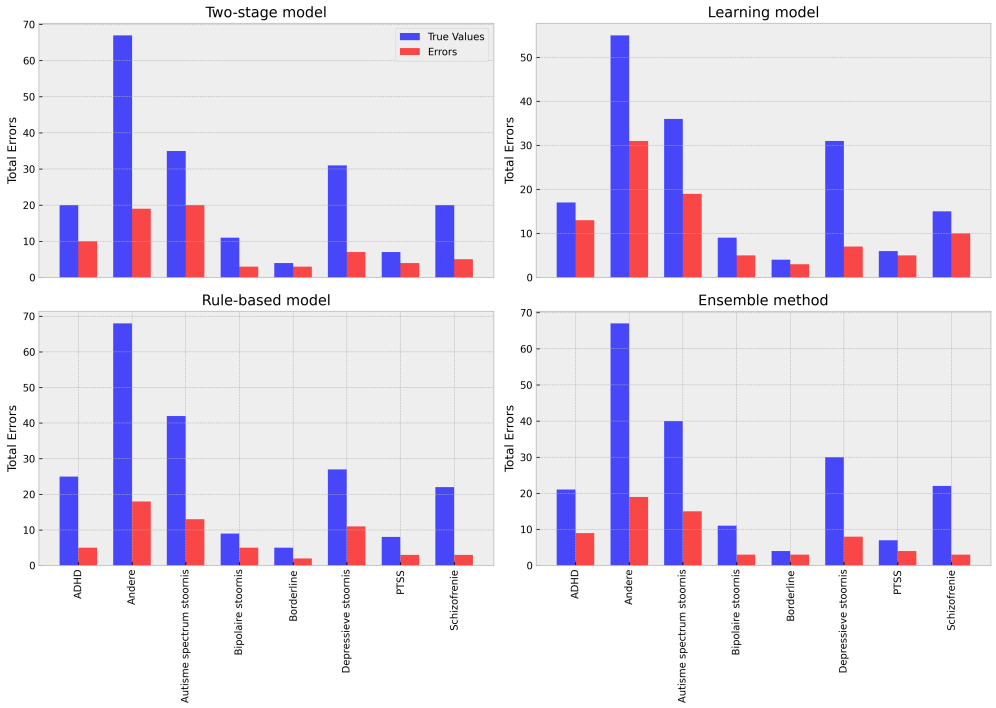

Figure 14: Amount of errors made by all models stratified on disorder category

3.2 Error analysis

Apart from the misclassifications as seen in Figure 13, which made apparent that all models prefer to classify diagnosed, several errors persist across all methods. These errors highlight aspects that account for most of the misclassifications. Apart from errors that fall in the category of being inherent to the models and out of the scope, for example: mentioning that a disorder is not relevant in sentence no. 2, and only mentioning the disorder in sentence no. 6, -or-, capturing every possible form of negation as defined rules, which is simply not possible given all combinations and possible misspellings. Other categories might be able to be handled in future versions of these models.

Fictitious examples of errors that persisted and can be categorized as being out of the scope or handleable are as follows:

The concatenation category highlights errors that have occurred due to multiple disorder terms being concatenated on one flag term. In the example below, all disorders should be negated due to the flag term 'geen aanwijzingen voor', but instead, the first disorder term gets negated and the flag term gets terminated. The rest of the disorder terms are therefore incorrectly not negated.

Gebaseerd op wat wij tot nu toe hebben gezien, zien we op dit moment geen aanwijzingen om een STRT{ontwikkelingsstoornis}ANDInvt (STRT{ADHD}ADHDnvt en/of STRT{autisme}ASSnvt) vast te stellen.

The same applies to the next example, where 'trekken' is a flag term that implies a consideration for both mentioned disorders, but in practice only gets applied to the first disorder term.

> Tijdens onderzoek trekken gezien die zowel bij STRT{ASS}ASSo als STRT{sociale angst}ANDIo kunnen passen, doorverwezen voor nadere diagnostiek.

The ContextD algorithm suffers most from this way of writing. In this algorithm, the scope is terminated when one rule is applied. The other models should make sense from these outcomes by masking other disorders.

Another type of error category is how the models handle disorders mentioned in terms of treatment centers. When disorders are mentioned in terms of treatment centers, these should be negated, since these disorders are not explicitly diagnosed or considered. Both of these examples show disorders mentioned in terms of treatment centers, and all models got these wrong. As of writing this paper, no flags were implemented to highlight these terms that the models could learn from. The models see these as normal words, leading to misclassifications.

> Behandeling gaat door bij INSTELLING-1 STRT{bipolair}BIP1nvt, en wordt afgesloten in het INSTELLING-2.

> Op de poli STRT{persoonlijkheidsstoornissen}ANDInvt aangemeld in verband met stemmingswisselingen en hypomane episodes.

The category of flag terms that are not properly recognized is the biggest error category, but, as stated above in the introduction of this section, cannot be eliminated. However, expanding on rules can help improve the models that rely on the flagging of terms. In the example below, a simple case is given of a flagged term, namely: 'onvoldoende aanwijzingen gevonden voor', that should be recognized, but because only variations (e.g. 'onvoldoende aanwijzingen voor') are present the rule-set, the flagged term is not captured correctly. The problem with looking at errors and mining rules in such a manner is that for every new instance, this algorithm will be used, and new rules will need to be mined.

> Uit de voorgeschiedenis werden onvoldoende aanwijzingen gevonden voor STRT{autismespectrumstoornis}ASSnvt.

The last category pertains to disorders being mentioned in terms of medication. The reason this is misclassified is that it is not explicitly obvious that the disorder has been diagnosed, considered, or ruled out. Instead, the disorder is mentioned in terms of medication and models will pick this up as a diagnosis. This is the case in the example below. In this example, the disorder is mentioned in terms of being the cause for giving medication, but this is not an explicit diagnosis in the way the texts were annotated.

> Geadviseerd om te stoppen met andere middelen en aanzet om STRT{ADHD}ADHDnvt klachten te mitigeren met Methylfenidaat.

4 Conclusion and discussion

This project acts as the foundation for making the pivot towards quantifying disorders and further analyzing large amounts of patient data within the Psychiatry department of the UMC Utrecht. Along with the other sub-projects, this sub-project tested a completely rule-based approach, a partly rule-based and deep learning model, and a two-stage model classifying scope vectors using rule-based term flagging, on 500 randomly selected outpatient- and discharge letters. The results show that the least complex system performed the best after validating. Inherent transparency within these less complex models adds value in the way of interpretability towards specialists. This is the same for the unilateral processes across the sub-projects and model-independent processes for concept extraction and evaluation. These steps are pragmatic and transparent. Performance across all models is subpar when compared to referred- and original literature. Application of these systems is fairly straightforward but the downstream tasks can be optimized, as will be discussed in the discussion in section 4.1.

4.1 Discussion

Investigating the different models gave all sorts of insights into the texts, such as errors persisting even after a heterogeneous approach to classifying disorders. Having obtained a precision of 0.94, a recall of 0.88, and an F1-score of 0.91 on the consensus set classified on the consensus set (considered to be the ground truth) [van Ginkel 2024], this inference seems to be largely correct. Tuning the models to the texts also gave insight into the best approach to handle these texts, and the results showed what worked better on these texts than what is otherwise considered to be the default. Having a larger scope worked better for these types of texts, suggesting that the context is spread out over more words than compared to different (healthcare) texts.

The entirely rule-based system with supplemented rules and the ContextD algorithm achieved an accuracy of 0.78, an unweighted recall of 0.72, and an F1-score of 0.77. The majority vote ensemble model had the highest precision: 0.78. The two-stage model, using flagging ContextD terms, part-of-speech tagging, and classifying via SVM, performed third best overall. The learning model using MedRoBERTa transformers to classify the disorder performed worst overall. In general, the models could in theory perform much better, however there are too many factors that are missed in classifying. For the more rule-based models these factors are terms or phrases that are not mapped correctly, and for the transformer model this would (most likely) be a class imbalance for psychiatric data, leading to bad performance.

Reflecting on the literature for reference, van Es et al. (2023) reported an F1-score of 0.86 for the rule-based approach also using ContextD, and an F1-score of 0.95 for the transformer-based model (the same as used in this report). Going from the results, they concluded that overall, the learning model performed best, and for structured texts, the rule-based system could keep up. Results from this project do not necessarily align with these findings but do suggest that the rule-based approach is more adaptable than the transformer set up by van Es et al. (2023). The same goes for the research done by Goryachev et al. (2007), where the two-stage mode originated. In their paper, the authors pitch the two-stage model against the precursor of ContextD, NegEx. Following the performance metrics, (their models portrayed an F1-score of 0.91 and 0.86 for the NegEx and the two-stage respectively), they concluded that the classification-based system performed worse than the "regular expression and syntactic

processing-based algorithms". These findings align with the findings in this report, where the rule-based model also outperformed the two-stage model.

4.1.1 Model discussion

Each model benefits from different changes and supplementation. This section highlights what the best course of action would be to directly advance these models for performance.

Rule-based model The rule-based model was the best-performing model overall, beating every other model except for the ensemble model concerning precision. The unweighted recall is relatively high compared to the rest of the models, suggesting that this approach performs best across all classes. It appears that the ContextD algorithm captures the context sufficiently, especially when compared to the rest of the models. The added rules are not syntactically different, but rather variations of the spelling for the already existing rules. Expanding on this could result in further improved performance.

Learning model It is clear from the performance of the learning model that it has trouble making the pivot toward pure psychiatric text. Given the fact that this transformer was trained on clinical text, where apparently, not a significant amount of similar syntactic text is present. Due to the black-box nature of this model, the exact workings and therefore deficiencies, could not be uncovered. The probability of classifying 'Diagnosed' is high at default (see the thresholds during the tuning in Figure 9), and moving this threshold higher to let more classifications fall into the 'Considered' class, does not move the correct classifications into the next class. Adjusting both these thresholds higher pushes all classifications into the 'NA' class. In conclusion, the model does not capture the correct context and/or aspects of the text to make the right classifications.

It could be that the rule-based aspects of this approach, such as splitting the sentences or matching the entities based on a list, go against what the model performs best on, namely raw text. But then again, if raw text were presented, it would not be much of a comparison when looking at: the other models in this report, and the other sub-project that researches LLMs on these texts.

Two-stage model The two-stage model shows promise because it can combine the rules that directly capture context, into a more generalizable format due to the part-of-speech tagging. This means that a lot of different classifiers can in theory more easily find complex relations in this dense data space. However, being two-stage, every stage brings about possible bias. The part-of-speech tagging may not be optimal, or the tagging of the rules. The problem this model suffers from now is that when a rule is not flagged, the subsequent classifier finds nothing to relate to. This explains why this model could not perform better than the only rule-based model since this model at least has a default to classify 'Diagnosed'.

Ensemble model The ensemble model predicts the class that two or more votes agree on. These votes are built up by the stacked classifiers, namely all of the above. The goal is to balance out the shortcomings of respective models with models that can supplement these shortcomings. This concept worked out to be beneficial, especially since the ensemble model performed better than two of the models it was stacked with. The learning model strictly classifies 'Diagnosis' classes, while the other models try to ascertain if the other classes are the

case. This approach gives the highest weighted precision but still suffers from low recall due to the misclassifications of the learning model.

4.1.2 General discussion

Model deficiencies have been highlighted in paragraph 3.2. In summary, all models benefit from handling concatenation correctly, this occurs when one trigger term applies to multiple disorders. The same goes for handling mentions of disorders in terms of medication and treatment centers correctly. Annotation was done in such a way that explicit mentions of disorder status were definitive. Implicit mentions of medication due to a disorder imply causality and are not correctly handled. In general, expanding on the explicit trigger terms used in psychiatric care can greatly improve the performance of these types of texts, and give insight into the differences between psychiatric texts and different types of healthcare.

General recommendations for improving the overall effectiveness of the project in the long term are as follows: tracking disorder evolution on an individual patient level and/or classifying sub-types of disorders within the text. The first recommendation could be achieved by tagging re-occurring disorders in notes and linking different types of letters. The second recommendation comes forth out of the top-level classifications focus concerning the disorder categories. However, sub-types within these categories can be mentioned differently and therefore be missed in evaluation. For example, think of developmental disorders that are supposed to be part of the autism spectrum, but are mentioned differently and more vaguely. The next step could be to analyze causal symptoms in text related to the disorder. This could give insight into all different kinds of new information. To test these eventual implementations, a greater emphasis on external validation could highlight weak points and make the setup more robust, but looking for a between set to validate can be quite a challenge.

5 References

Afzal, Z, E Pons, N Kang, MC Sturkenboom, MJ Schuemie, and Kors JA (2014). "ContextD: an algorithm to identify contextual properties of medical terms in a Dutch clinical corpus". In: *BMC Bioinformatics*.

— ((2014)). "ContextD: an algorithm to identify contextual properties of medical terms in a Dutch clinical corpus". In: *BMC Bioinformatics*.

Berge, G.T., OC. Granmo, T.O. Tveit, and et al. (2023). "Combining unsupervised, supervised and rule-based learning: the case of detecting patient allergies in electronic health records". In: *BMC Medical Informatics and Decision Making*.

Chapman, W.W., W. Bridewell, P. Hanbury, G.F. Cooper, and B.G. Buchanan ((2001)). "A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries". In: *Journal of Biomedical Informatics*.

Goryachev, Sergey, Margarita Sordo, Qing T. Zeng, and Long H. Ngo (2007). "Implementation and Evaluation of Four Different Methods of Negation Detection". In.

— ((2007)). "Implementation and Evaluation of Four Different Methods of Negation Detection". In.

HGR (2019). "DSM(5): Toepassing en status van de diagnose en classificatie van geestelijke gezondheidsproblemen". In: *Hoge Gezondheidsraad* 9360.

Hoek, M (2024). "Extracting diagnoses from Dutch psychiatric text using pre-trained Large Language Models". In: URL: `https://studenttheses.uu.nl/discover?field=studyProgramme&query=applied%20data%20science&filtertype=studyProgramme&filter_relational_operator=equals&filter=Applied+Data+Science`.

Honnibal, Matthew, Ines Montani, Sofie Van Landeghem, and Adriane Boyd (2020). "spaCy: Industrial-strength Natural Language Processing in Python". In: DOI: `10.5281/zenodo.1212303`.

Le Glaz, A., Y. Haralambous, D.H. Kim-Dufor, P. Lenca, R. Billot, T.C. Ryan, J. Marsh, J. DeVylder, M. Walter, S. Berrouiguet, and C. Lemey ((2021)). "Machine Learning and Natural Language Processing in Mental Health: Systematic Review". In: *J Med Internet Res*.

Malgaroli, M., T.D. Hull, J.M. Zech, and et al. ((2023)). "Natural language processing for mental health interventions: a systematic review and research framework". In: *Translational Psychiatry*.

Menger, V, B van Es, and mcsnakey (2024). "umcu/clinlp: v0.8.1". In: URL: `https://github.com/umcu/clinlp`.

Menger, V., F. Scheepers, L.M. van Wijk, and M. Spruijt (2018). "DEDUCE: A pattern matching method for automatic de-identification of Dutch medical text". In: *Telematics and Informatics* 35.4, pp. 727–736. ISSN: 0736-5853. DOI: `https://doi.org/10.1016/j.tele.2017.08.002`. URL: `https://www.sciencedirect.com/science/article/pii/S0736585316307365`.

OpenAI ((2024)). "ChatGPT 4.0 [Large language model]". In: URL: `https://chatgpt.com/`.

Pedregosa, F., 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 (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Pies, Ronald (2007). "How "Objective" Are Psychiatric Diagnoses?" In: *Psychiatry (Edgmont)*.

Rezaii, N., E. Walker, and P.A. Wolff ((2019)). "A machine learning approach to predicting psychosis using semantic density and latent content analysis". In: *npj Schizophrenia*.

Rezaii, N., P. Wolff, and B.H. Price ((2022)). "Natural language processing in psychiatry: the promises and perils of a transformative approach". In: *The British Journal of Psychiatry*.

Swaminathan, A., I. López, R.A.G. Mar, and et al. ((2023)). "Natural language processing system for rapid detection and intervention of mental health crisis chat messages". In: *npj Digital Medicine*.

van Es, B., L.C. Reteig, S.C. Tan, and et al. ((2023)). "Negation detection in Dutch clinical texts: an evaluation of rule-based and machine learning methods". In: *BMC Bioinformatics*.

van Ginkel, E (2024). "Annotation of Psychiatric Data". In: URL: `https://studenttheses.uu.nl/discover?field=studyProgramme&query=applied%20data%20science&filtertype=studyProgramme&filter_relational_operator=equals&filter=Applied+Data+Science`.

Velupillai, S, H Suominen, M Liakata, A Roberts, AD Shah, K Morley, D Osborn, J Hayes, R Stewart, J Downs, W Chapman, and R Dutta (2018). "Using clinical Natural Language Processing for health outcomes research: Overview and actionable suggestions for future advances". In: *Journal of Biomedical Informatics*.

— ((2018)). "Using clinical Natural Language Processing for health outcomes research: Overview and actionable suggestions for future advances". In: *Journal of Biomedical Informatics*.

Verkijk, S. and P. Vossen (2021). "MedRoBERTa.nl: A Language Model for Dutch Electronic Health Records". In: *Computational Linguistics in the Netherlands Journal*.

Appendices

A Glossary

This glossary is made possible thanks to ChatGPT, courtesy of OpenAI (2024).

Term	Definition
Algorithm	A set of rules or steps to be followed in calculations or problem-solving operations.
Association Rules	Rules that highlight relationships between variables in large datasets.
BERT	Bidirectional Encoder Representations from Transformers: a transformer-based model designed to pre-train deep bidirectional representations by jointly conditioning on both left and right context in all layers.
biLSTM	Bidirectional Long Short-Term Memory: a type of recurrent neural network that can capture context from both directions in a sequence.
Classifiers	Algorithms used to categorize data into predefined classes, such as SVM, logistic regression, and neural networks.
Convolutional Neural Networks	A class of deep neural networks, most commonly applied to analyzing visual imagery.
ContextD	An algorithm designed to replace NegEx for better detection of negation and context in clinical text.
Contextual Encoder Representations	Representations like BERT that understand the context of a word in a sentence.
Deep Learning	A subset of machine learning involving neural networks with many layers, enabling the processing of complex data patterns.
Decision Trees	A decision support tool that uses a tree-like model of decisions and their possible consequences.
Domain Knowledge Features	Features derived from specific domain knowledge, such as UMLS and DSM-X.
ELMo	Embeddings from Language Models: a deep contextualized word representation that models both complex characteristics of word use and how these uses vary across linguistic contexts.
Electronic Health Records (EHRs)	Digital versions of patients' paper charts that contain medical history, diagnoses, medications, treatment plans, and test results.
Embedding	A representation of text in continuous vector space, such as word2vec.
Experiencer	The entity that experiences a certain condition or event in a clinical context.
GloVe	Global Vectors for Word Representation: an unsupervised learning algorithm for obtaining vector representations for words.
HTML	HyperText Markup Language: the standard language for creating web pages and web applications.
Hybrid-based Models	Models that combine different architectures or techniques to improve performance.

LDA	Latent Dirichlet Allocation: a generative statistical model used for topic modeling.
LIWC	Linguistic Inquiry and Word Count: a text analysis software that calculates the degree to which people use different categories of words.
LLMs	Large Language Models: models with a large number of parameters that are trained on extensive text datasets to understand and generate human-like language.
LSA	Latent Semantic Analysis: a technique in natural language processing for analyzing relationships between a set of documents and the terms they contain.
Lemmatization	The process of reducing words to their base or root form.
Linear Regression	A linear approach to modeling the relationship between a dependent variable and one or more independent variables.
Linguistic Features	Features based on linguistic properties, such as part-of-speech tagging and bag of words.
Logistic Regression	A statistical model used for binary classification tasks.
Machine Learning	A subset of AI that involves the use of algorithms and statistical models to enable computers to improve at tasks with experience.
MedCAT	Medical Concept Annotation Tool: a tool for annotating medical concepts in text using natural language processing.
Mined	The process of extracting useful information from large datasets.
N-grams	Contiguous sequences of n items from a given sample of text or speech.
NLP	Natural Language Processing: a field of AI focused on the interaction between computers and human language.
NER	Named Entity Recognition: the process of identifying and classifying entities in text into predefined categories.
Naive Bayes	A classification technique based on Bayes' theorem with an assumption of independence among predictors.
NegEx	An algorithm used to detect negation in clinical text.
Negation	The grammatical construction that contradicts or denies some or all of the meaning of a sentence.
Neural Network	A series of algorithms that attempt to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.
POS	Part-Of-Speech tagging: the process of marking up words in a text as corresponding to a particular part of speech.
Preprocessing Techniques	Techniques used to prepare raw data for analysis, including lemmatization, POS tagging, and n-grams.
Random Forest	An ensemble learning method that operates by constructing multiple decision trees.
Recurrent Neural Networks	A class of neural networks where connections between nodes form a directed graph along a temporal sequence.
RobBERT	A transformer-based model specifically designed for processing Dutch language text.
SVM	Support Vector Machine: a supervised machine learning algorithm used for classification and regression tasks.

Statistical Features	Features extracted from text using statistical methods, such as TF-IDF and n-grams.
TF-IDF	Term Frequency-Inverse Document Frequency: a statistical measure used to evaluate the importance of a word in a document relative to a corpus.
Temporality	The aspect of time concerning when a medical event occurred.
Thesaurus Linkage	Associating words with their synonyms using a thesaurus.
Transformer-based Models	Models that rely on transformer architecture, which is particularly effective for NLP tasks.
Trigger and/or Flag Terms	Words or sets of words representing a status of a disorder, like indicating a diagnosis, a consideration or that the disorder is not applicable.
UMCU	University Medical Center Utrecht.
UTF-8 Characters	A standard for encoding a wide array of characters in the Unicode standard using 8-bit blocks.
Word2vec	A group of related models that are used to produce word embeddings.

B Repository

Upon request, the code for the project and its dependencies can be obtained. Note that this does not include the data, since this is protected by privacy regulations. The structure of the repository is as follows:

```
⟼ DevOps
    ↪ Dependencies
    |   ↪ specifications.txt
    |   ↪ package_modifications.txt
    ↪ Pre-processing
    |   ↪ entity_and_annotation_mining.ipynb
    |   ↪ letter_sampling_conclusion_filtering.ipynb
    |   ↪ exploratory_data_analysis.ipynb
    ↪ Validating
    |   ↪ rule_based_cv.ipynb
    |   ↪ learning_model_cv.ipynb
    |   ↪ two_stage_cv.ipynb
    ↪ Main
    |   ↪ rule_based_main.ipynb
    |   ↪ learning_model_main.ipynb
    |   ↪ two_stage_main.ipynb
    |   ↪ ensemble_model_main.ipynb
    ↪ Evaluation
        ↪ prediction_matrix_error_analysis.ipynb
```