Language as a Predictor of Anxiety, Depression, and Self-Efficacy Scores and Recovery Rate in Teenagers with Chronic Fatigue Syndrome

Mara Fennema

7021461

Supervisors:

Tejaswini Deoskar

Marijana Marelj

Master Artificial Intelligence

Utrecht University

October 28, 2022

Contents

Abstract

Nowadays, Artificial Intelligence (AI) models are being used in multiple areas of the healthcare sector. This thesis looks into the relationship between language use of teenaged patients with Chronic Fatigue Syndrome (CFS) and their anxiety, depression, self-efficacy, and CFS treatment outcome. This research aims to make it easier for healthcare professionals to get an indication of the level of a patient's anxiety or depression, the measure of their self-efficacy, and whether or not a specific type of treatment will work for a patient. Using a short text written by the patient to get such an indication would facilitate an earlier start of effective treatment. This thesis uses data from 102 patients who received online email-based Cognitive Behavioural Therapy for its two main focus areas. The first focus area looks at the correlation between a patient's language use and their anxiety, depression, and self-efficacy. This is done by training n-gram-based language models and Naive Bayes on the text in the emails to predict the patients' anxiety, depression, and self-efficacy scores. The language models' results were compared to those of models trained on randomly generated scores, and it was shown that outputs of these models were statistically significant. The language model performed better than Naive Bayes, and it was concluded that there was a correlation between language use and anxiety, depression, and self-efficacy. The second focus area looks at how well the language used by the patients in the emails sent to their therapists can be used with various AI models to predict the level of their anxiety and depression, the measure of their self-efficacy, and their CFS treatment outcome. This was done using the number of non-agentic language features per email, Bag of Words, and BERTje embeddings. These features were used as input for both logistic regression models and neural networks. When using logistic regression, the models for predicting self-efficacy using BERTje embeddings performed best. The neural networks using BERTje embeddings outperformed the logistic regression models when predicting anxiety, depression, self-efficacy, and treatment outcome. Thus it was concluded that it is possible to predict anxiety, depression, self-efficacy, and patient recovery based on language use.

Acknowledgements

Firstly, I would like to thank Tejaswini Deoskar and Marijana Marelj for their support as my supervisors during this thesis.

Secondly, thanks to Sanne Nijhof and Elise van de Putte, without their data from the FITNET system they have developed, this research would not have been possible. Additionally, meeting with them and talking about what use cases for the results of this thesis were possible was very valuable for the process.

Thirdly, I would like to thank Tessa Wagenaar for her continued support and brainstorming sessions when I got stuck with this research.

Additionally, I would like to thank Ghislaine van den Boogerd and Noa Visser for their support and willingness to discuss this thesis when necessary.

Moreover, a big thanks to Liesbeth Jongkind, Fokke Fennema, Naomi Spaans, Anneloes van Schaik, Claire Paulissen, Femke Reniers, Tessa Wagenaar, Ghislaine van den Boogerd, Noa Visser, and Floor van de Leur for proofreading parts of this thesis during the writing process.

Lastly, a form of gratitude should be extended to the fact that I was diagnosed with Chronic Fatigue Syndrome (CFS) as a teenager. Through having CFS from when I was 15 until I was 17, the subject of this thesis was very personal, which helped my motivation immensely.

1 Introduction

Since the rapid development of Artificial Intelligence (AI) in the past decades, more and more AI-based solutions have been created to help decrease the workload for healthcare workers and speed up diagnosis. Many of these applications have been computer vision-based, for example, focusing on detecting cancerous cells in images (Al-shamasneh and Obaidellah, 2017; Kohlberger et al., 2019; Wu et al., 2019). In certain other sub-fields of healthcare, such as in ophthalmology and pathology, AI models have even reached human-level performance (Ehteshami Bejnordi et al., 2017; Gulshan et al., 2016).

Not only has research using AI been done with a focus on the physical aspects of health, but also with a focus on mental health. Such research can be used to aid or accelerate diagnosis of mental health conditions. Diagnosis and treatment of mental health-related issues are generally done in the Netherlands by an institution affiliated with the GGZ. For diagnosis and start of treatment, these institutions are to follow the Treeknorm GGZ, the maximally acceptable waiting time of 14 weeks until the beginning of treatment, which was decided in 2000 (GGZ, 2019). In 2019, 92% of standard mental health clinics would make the Treeknorm GGZ, while only 73% of specialised clinics would (GGZ, 2019). Through AI applications, more ways to indicate diagnosis than the standard routes might be possible. This thesis looks at ways to determine the state of a patient's anxiety and depression and the measure of their self-efficacy, as well as ways to predict whether the patient recovers from Chronic Fatigue Syndrome based on their language use. By using language use as an indicator of the state of one's mental health, a diagnosis might happen faster and requires less effort from the therapists. This way, the waiting times could be decreased, and thus more mental health clinics will be able to make the Treeknorm GGZ. By having shorter waiting times, a patient's treatment can start earlier, which would be good for their mental health.

1.1 Introduction Chronic Fatigue Syndrome

Chronic Fatigue Syndrome (CFS) is an ailment characterised by severe disabling fatigue for at least six months, combined with symptoms ranging from concentration impairments and bad short-term memory to musculoskeletal pain and sleep disturbances (Fukuda, 1994; Sharpe, M. C. et al., 1991). Due to these symptoms, patients suffering from CFS experience significant functional impairment (Afari and Buchwald, 2003). Examples of such impairments are a decrease in social relationships and lower attendance at school or work (Afari and Buchwald, 2003). The majority of people who are diagnosed with CFS are 30-40 years of age (Afari and Buchwald, 2003). However, adolescents can also contract the disease, and it is estimated that about 0.11% of adolescents suffer from CFS (Nijhof et al., 2011).

Adolescents with CFS are heavily impacted by their syndrome, as almost all patients report missing school more than 50% of the time in the last six months due to CFS (Nijhof et al., 2011). Peer interaction at places such as school as an adolescent is a vital aspect of development, and increased social isolation can negatively affect mental health (Orben et al., 2020). Adolescents suffering from chronic disease report lower life satisfaction and psychosomatic health and more mental health problems than their healthy peers (Berkelbach van der Sprenkel et al., 2021). These mental health problems are known to increase impairments in adolescents, such as social impairment in areas of home life, friendships, classroom learning, and leisure activities (Wille et al., 2008).

This thesis will look at two mental health disorders, these being anxiety and depression, along with self-efficacy. Anxiety disorders are defined as clinically significant, unpleasant emotions that have the quality of dread, fear, and alarm (Bouras and Holt, 2007). Depression is a mental disorder generally characterised by a loss of feeling pleasure from activities that would usually bring joy due to a disturbance in the brain's neurotransmitters (Gilbert, 2006). Self-efficacy is defined as an individual's belief in their capacity to execute behaviours required to produce specific performance goals (Bandura, 1977). This thesis will look further into the relationship between anxiety, depression, and self-efficacy and the language use of the patients suffering from them.

1.2 Introduction Subjective and Objective Data

This thesis uses data from Nijhof et al. (2012). They researched a new treatment for teenagers suffering from CFS, called Fatigue In Teenagers on the interNET, hence FITNET. FITNET is an online platform where patients received a form of Cognitive Behavioural Therapy (CBT) via email. As these emails were the only form of CBT therapy these patients received, these emails can be seen as transcriptions of their therapy sessions. This thesis uses these emails, along with other data collected from these patients. This other data consists of the answers patients gave to self-report questionnaires regarding anxiety, depression, and self-efficacy. A more in-depth description of the data itself can be found in Chapter 2.

This thesis introduces a distinction between two parts of the data used. These two parts are henceforth referred to as the subjective data and the objective data. The subjective data consists of the answers they gave on self-report questionnaires. The objective data consists of emails the CFS patients sent to their therapists.

For the purposes of this research, the subjective data is called subjective because the patients are explicitly asked about their symptoms, which might influence their answers. When they are asked, for example, how often each week they feel exhausted, they will actively think about their symptoms. This can make it seem worse for the patients, thus influencing their answers. Additionally, they know a doctor will read their answers and most likely will take them into account to devise a treatment plan. This, too, can influence how they answer the questions, as they can exaggerate their symptoms in the hopes of getting treatment faster, or they can downplay them if they do not want to make it sound too bad.

The objective data, however, is called objective as the patients are not as aware of all of this. As the patients are asked to describe their siblings and friends, not all the emails they send are directly about their symptoms. Because of this, this research calls these emails objective data, as the patients are less aware of their symptoms, and they are also less likely to feel judged, as the emails they send are not direct tests or surveys. Telling a therapist the name of your siblings is different from telling a therapist how often you feel anxious.

1.3 Previous Research

Dalmaijer et al. (2021) looked at the recovery rate of CFS patients in relation to their language use, focusing specifically on their use of Non-Agentic Language. Non-agentic language is a way of removing the agent from a sentence, for example, '*The vase got broken*' as opposed to the sentence '*I broke the vase*'. Dalmaijer et al. (2021) found that patients who did not recover from CFS used non-agentic language significantly more often than patients who did recover. Additionally, based on the same data, higher percentages of use of linguistic indicators of catastrophizing were found to be related to poorer treatment outcomes (Wignand, 2021). An example of catastropizing would be, 'If I don't get better quickly, I will never get better, and I will have CFS for the rest of my life.' This thesis uses the same data as was used by Dalmaijer et al. (2021) and Wignand (2021). This data set was a byproduct of the research done by Nijhof et al. (2012).

Not only has it been shown that non-agentic language is indicative of patient recovery but also of mental health conditions. Oren et al. (2016) showed that people with Obsessive Compulsive Disorder (OCD) are more prone to use such language compared to people without OCD.

Other linguistic markers have also been found to appear more prominently in the language use of people with other mental or physical health diagnoses. For example, words denoting totality, either of magnitude or probability, such as ‘totally’ or ‘completely’, were found to be more commonly used on forums relating to anxiety, depression, and suicidal ideation than on other forums (Al-Mosaiwi and Johnstone, 2018). Catastrophizing was also found to be significantly associated with momentary fatigue in CFS patients (Sohl and Friedberg, 2008). Aside from catastrophizing, depression and anxiety symptoms were also found to be significantly associated with this momentary fatigue.

Using AI, the relationship between other mental health conditions and language used by patients has been researched as well. For example, it has been shown that based only on text written by patients, the severity of their depressive symptoms can be predicted better than by other models so far (Hong et al., 2022). Moreover, Coppersmith et al. (2015) looked into predicting different self-diagnosed mental health diagnoses from Twitter data. They found that there are possible markers for a multitude of diagnoses, which implied that further research could find clearer ones. Bucur et al. (2021) found a possible example of such a feature, as they found that patients suffering from depression more frequently use offensive language than people without depression.

1.4 Broad Methodology for this Research

The data used for this thesis contains information about teenagers diagnosed with CFS. The reason this study focuses on adolescent CFS patients is because Nijhof et al. (2012) created the only data that allows for comparison between language use, the state of a patient's anxiety, depression, and the measure of their self-efficacy, and whether the patient recovered from their CFS. This is because this data consists of two parts, the subjective and the objective data, as is described in Section 1.2.

By comparing the transcriptions of therapy sessions from the objective data with the mental health scores from the subjective data, possible correlations between language use and mental health diagnoses can be found. If such correlations between linguistic features and mental health diagnoses exist, these indicators could be used to aid diagnosis.

Additionally, the second part of this thesis also looks at predicting whether patients recover using a certain type of healthcare program. This could improve the treatment experience for the patient. Patients would, with accurate prediction, be less likely to receive treatment that would not result in them recovering. If done correctly, less care would be necessary, as fewer patients would need multiple types of treatment in the hopes of one of them working. This would alleviate pressure on the healthcare system. If another situation might occur where hospitals have to postpone or cancel certain appointments, as happened due to COVID-19 (Azam et al., 2020; Slotman et al., 2022), having fewer appointments in total could allow for fewer treatments being cancelled.

This research aims to answer two questions. The first is '*Is there a correlation between the language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy and the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires?*'. The hypothesis is that there is a correlation between language use and the level of a patient's anxiety and depression, and the measure of their self-efficacy (Al-Mosaiwi and Johnstone, 2018; Bucur et al., 2021; Coppersmith et al., 2015; Hong et al., 2022; Oren et al., 2016).

The second research question is '*To which extent can language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy be used* *by AI models to predict the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires, and whether they will recover from CFS through this therapy?'* It is expected that the anxiety, depression, self-efficacy, and the outcome of therapy can be predicted to an extent, limited by the size of the data set. For anxiety, depression, and self-efficacy, this is based on the AI research discussed before, which showed that text written by people can be used to determine whether someone has depression, or determine the severity of their depressive symptoms (Bucur et al., 2021; Hong et al., 2022). As for using text to predict the recovery rates, this is based on the fact that non-agentic language and catastrophizing can be key indicators of whether a patient recovers from CFS (Dalmaijer et al., 2021; Wignand, 2021).

The relevance of this research for the field of AI is that it aims to create a way to help medical research using an AI application. As this is a very practical application, the field of AI might become more understandable, as opposed to the abstract field it is to a lot of people. As this research also combines linguistics, medicine, and AI, the interdisciplinary research fields get extended.

The rest of this thesis is constructed as follows. Chapter 2 describes both the contents of the subjective and objective data set, as well as the pre-processing steps taken for each. Additionally, it also explains some measures taken to lessen the influence of the data set being relatively small for an AI application. In Chapter 3, I discuss the part of the research regarding the first research question. Here, two approaches are explained, one based on the subjective data and one on the objective data. The methods for both approaches are discussed, as well as the results for each. Lastly, the conclusions based on these two approaches are shown. Chapter 4 deals with the second research question. Here, the methods of how the emails from the objective data have been represented by numerical data are explained, as well as all the models used. Subsequently, the results are discussed, followed by the conclusions. Chapter 5 concludes the thesis.

2 Description of the Data Set

This thesis makes a distinction between subjective and objective data. The subjective data consists of answers given by teenagers with CFS on self-report questionnaires. The objective data consists of the emails these same patients sent to their therapists as part of their treatment. This chapter will describe the contents of both of these data sets in more detail. Both data sets used come from Nijhof et al. (2012), as these were the results of that research.

Besides the contents, all the pre-processing steps taken will be explained. The data description and pre-processing steps for the subjective data are described in Section 2.1. The explanation for the objective data and the pre-processing steps taken for those can be found in Section 2.2. Additionally, as each data set is quite small for an AI application, methods to prevent overfitting were taken, and are explained in Section 2.3.

An overview of how the data was used in each part of the research can be found in Figure 1. It shows that the output values (i.e. the anxiety, depression, self-efficacy scores and the recovery rates) for all the AI models come from the subjective data, along with the survey answers that some AI models use as input. The emails from the objective data are transformed into multiple representations. These representations are then used as the input features for different AI models.

2.1 Subjective Data and Pre-Processing

The full subjective data set contains information about 135 different patients. Of these 135 patients, 33 have been ignored for the purposes of this research. This is because these 33 patients did not use FITNET (an online platform where teenagers with CFS get online Cognitive Behavioural Therapy) during their treatment. As these 33 patients did not use

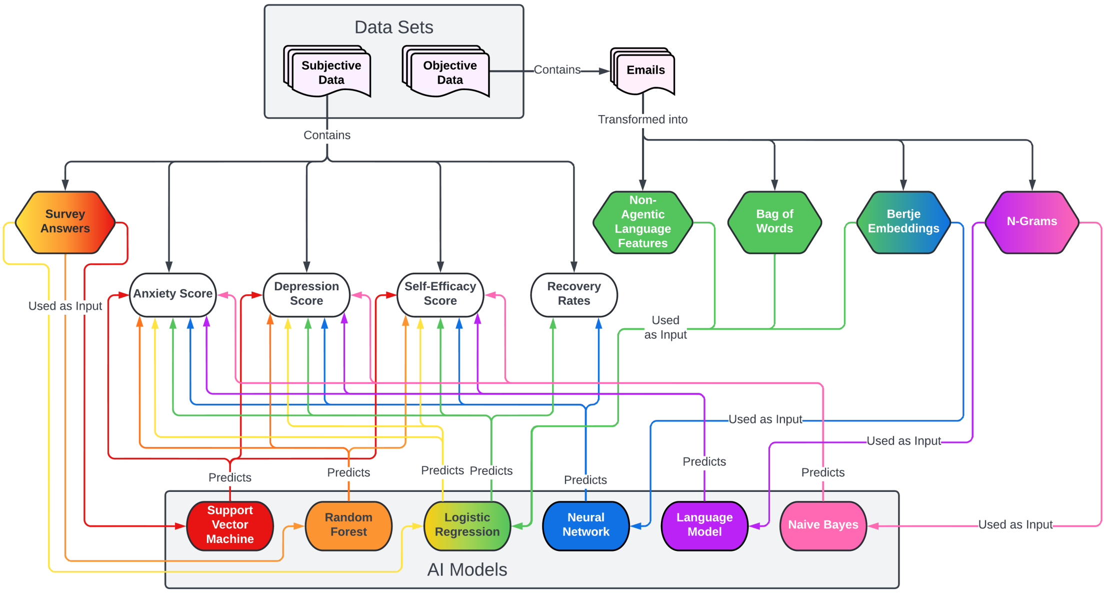


Figure 1: Flowchart overview of the created implementation. Shows the difference between the content of the subjective and objective data sets. Colours indicate the in and output of each different model. When a node in the flowchart consists of two or more different colours, that node's input and/or output differs based on the different coloured arrows. For example, logistic regression was used in two different ways. Once using the survey answers of the patients as input and anxiety, depression, and self-efficacy as output (shown in yellow). The other time, non-agentic language features, Bag of Words, or BERTje embeddings were used to predict anxiety, depression, self-efficacy *and* the CFS recovery rates (shown in green).

FITNET, no objective data from these patients exists. Only the data of the 102 patients of whom objective data exists has been used for the subjective data set.

So, the resulting subjective data set consists of anonymised data on 102 patients, with 152 features per patient. These features contain information ranging from how bad their CFS symptoms are, to how much school they are missing, psycho-social parameters like information on their parents and their mental health background and demographics such as gender, age, ethnicity, and school level for both the patients and their parents. A more in-depth explanation of the data can be found in Nijhof et al. (2012) and Dalmaijer et al. (2021).

To be able to use this data to train a predictive model, multiple pre-processing steps had to be taken. This is because the data had various features that would result in problems if they were not restructured or fixed. The issues were that the data was unstructured, had both missing values and outliers, and the categorical data had to be represented numerically. The solutions to these problems are described in the following sections.

2.1.1 Data Selection

As not all the data present in the full data set is relevant for this research, certain columns were ignored. Examples of such columns were ones with the name of the patient's therapist, their unique identification number, and whether or not they were treated using FITNET (FU) or Usual Care (UC). Usual Care could describe multiple things, including but not limited to Cognitive Behavioural Therapy or Group Therapy (Nijhof et al., 2012). As all the patients in this data set selection have used FITNET, this column was no longer relevant.

It is important to note that according to the information in this column, there were 35 patients who did not use FITNET. However, two of these 35 patients have sent emails through FITNET, thus objective data from them exists. Therefore, these patients were used as part of the subjective data set as well.

The original data set contains 152 columns, of which ten are ignored. This resulted in 142 columns containing data per patient. This is quite a lot, especially given the fact that there are only 102 patients in this data set, thus not all columns are used when training the model. This is because this is not enough data for the number of input features for any AI model.

The output values used by the AI models described in Chapters 3 and 4 are from the columns called **STAIC**, **CDI** and **CVSSE**. STAIC (State-Trait Anxiety Inventory for Children) is the score for anxiety disposition, CDI (Children's Depression Index) is the depression score, and CVSSE (Chronisch Vermoeidheids Syndroom Self-Efficacy) is the self-efficacy score. All the other columns can be used as input columns, and a selection can be quickly made between them.

To make it easier to experiment with including or excluding multiple columns at the same time, the columns were grouped together in categories. All the created categories, their descriptions, and their sizes are visible in Table 1. The code is implemented so that the categories can be quickly selected or deselected so that new models using different data can easily be trained.

2.1.2 Handling Missing Data

In every row in the data, at least one value is missing. This means that simply dropping each row missing a value would result in having no data left. Thus, another way to deal with the missing data has been implemented.

As the data consists of both numerical and categorical data, two ways were implemented, each unique to the type of data. For the categorical data, missing values were

Category	Description	Size
CDC	Questions and answers about symptoms using scores from the CDC	27
Demographics	Demographic information regarding the patients, such as age or gender	5
Econsults	Information about the number of e-consults	12
Follow-up Data	Information regarding the state of the symptoms during the follow-up questionnaire	18
Parents	Any information regarding the parents of the patients	36
Recovery	Information regarding the patients' mindset on their recovery	8
School	Any information related to school, such as attendance scores	11
Symptom Meta	Information about the duration of CFS and how it started	2
Symptom Scores	Scores regarding the intensity of symptoms	5
Usual Care	Information about the Usual Care certain patients received	6

Table 1: Overview of all categories created with descriptions and sizes of each category

simply viewed as a separate category.

For the numerical data, however, a different approach was used. The various numerical values within the data fluctuate quite a bit depending on the column, and both positive and negative numbers are used in multiple instances. Thus, a single value for all columns to indicate that data was missing was not a useful approach. Therefore, per column, the mean value of said column was calculated, which was used to replace the missing data in that column. Thus, each column has its own specific filler value, as each column has its own specific mean value.

The mean value of each column was taken, and not the median value, because some of the columns had very large ranges. As there were less than 102 data points per column, the differences between the data points could be very large. Thus, the larger values were also represented by taking the mean, which would not always be the case with the median. At the same time, this was also a drawback of using the mean, as possible outliers were given more influence over the filler value.

An example of what the data looked like before these missing values were filled can be found in Table 2. Table 3 is the exact same table, but with the missing values filled in the way described above.

Table 2: Example of what the data would have looked like before the missing values were filled. The data in this table is created for this example, and is thus not part of the actual data set.

	What is your age?	**How many weeks have you had CFS?**	**How often do you feel anxious?**	**How did your CFS start?**
Patient 1	15.6	85		Gradual onset
Patient 2		22	Once a week	Post-infectious
Patient 3	17.2		Once a month	Gradual onset
Patient 4	14.3	43	Twice a week	

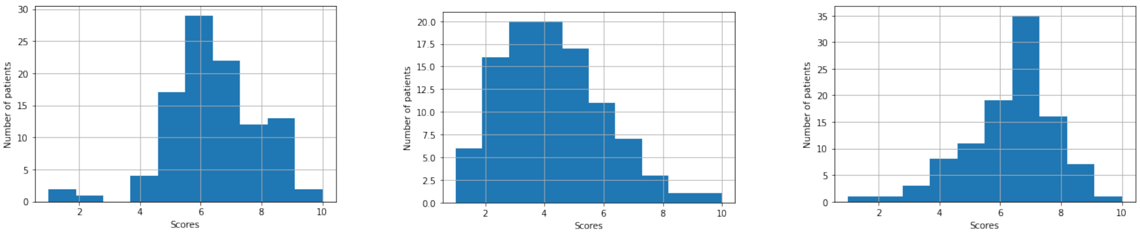
Table 3: Example of what the data looked like after the missing values were filled. The data in this table is created for this example, and is thus not part of the actual data set.

	What is your age?	**How many weeks have you had CFS?**	**How often do you feel anxious?**	**How did your CFS start?**
Patient 1	15.6	85	Unknown	Gradual onset
Patient 2	15.7	22	Once a week	Post-infectious
Patient 3	17.2	50	Once a month	Gradual onset
Patient 4	14.3	43	Twice a week	Unknown

2.1.3 Data Scaling

As mentioned above, the range of each column can vary greatly. Seeing as the data set only contains data on 102 patients, having values range in a single column from 0 to 102 makes it very hard for a model to make accurate calculations. Because of this, each numerical column, except for the output values, was scaled in such a way that all values were between 0 and 10 so that the values were closer together.

The output values, i.e. the anxiety, depression, and self-efficacy scores, are scaled as well, but between the values of 1 and 10. Additionally, after scaling, they are also rounded. By doing this, there are a set number of unique outputs possible, so that the model has a maximum amount of categories (i.e. 10). This makes it easier to have a higher accuracy score. In the original data, there are 33 different STAIC scores, ranging from 2 to 52 and



(a) Anxiety

(b) Depression

(c) Self-Efficacy

Figure 2: Histograms denoting the number of patients labelled with each value for all three mental health diagnoses for the subjective data. On the X-axis, the number denotes the score, and the number on the Y-axis denotes the number of times each value occurs in the data set.

33 unique CDI scores, ranging from 6 to 71. The CVSSE scores range between 8 and 24, containing a total of 16 unique values. Seeing as the classifiers used view the output as categorical, there are too many categories given the amount of data, as there are only 102 unique rows. Thus, by scaling the values to be within a certain range and proceeding to round them, there are significantly fewer categories, giving the model a bigger chance of getting the correct output. In Figure 2, the spread of the number of occurrences of each of the scaled categories is visible. Note that for anxiety, no data point falls in class 3.

2.1.4 One-Hot Encoding

As machine learning models require numerical data, the parts of the data that are categorical are transformed into numerical values using one-hot encoding, as this keeps the data unordered. An example of turning categorical data into numerical values, without keeping it unordered, would be to give one category the value 1, the second value 2, the third value 3, et cetera. While this makes the data numerical, it implies that the second categorical value follows the first, which is not true in the case of categorical data. If the categorical data denotes answers patients gave in a survey about whether or not they feel pain in their arm when performing a certain activity, the answers ‘Yes, in my right arm’, ‘Yes, in my left arm' or 'No' are not ordered values. Therefore, their numerical representation should be similarly unordered.

The solution to this is One-Hot Encoding. Instead of giving each possible answer a single number representation, they all get a vector. Each vector is of shape $1 \times M$, with M being the total number of categories in this column of the original data. So, for the example used before, there are three possible answers, thus three possible categories. Therefore, the vector for each answer is 1×3. For the first answer, let's say 'Yes, in my right arm', the vector would look as follows: $\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. For the answer, 'Yes, in my left arm', the vector would look like this: $\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$. And for the last answer, 'No', the vector would look like this: $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$. All of these vectors are numerical and unique, yet they do not imply any order in the possible answers.

However, due to the many zeros in the data, the size of the data becomes a lot bigger, making the data set a lot sparser as well. Sparser data can negatively influence the accuracy when training a model. However, sparseness cannot be prevented, as it is more important that the categories are independent than it is that sparseness is prevented.

2.2 Objective Data and Pre-Processing

As explained before, the objective data consists of emails, which can be viewed as transcriptions of their therapy as they were the only form of therapy that the patients received. Not every patient in the objective data set has used FITNET for the same amount of time. This is because some patients received usual care first. If their symptoms were still prevalent after usual care, the patients were given the option to also use FITNET.

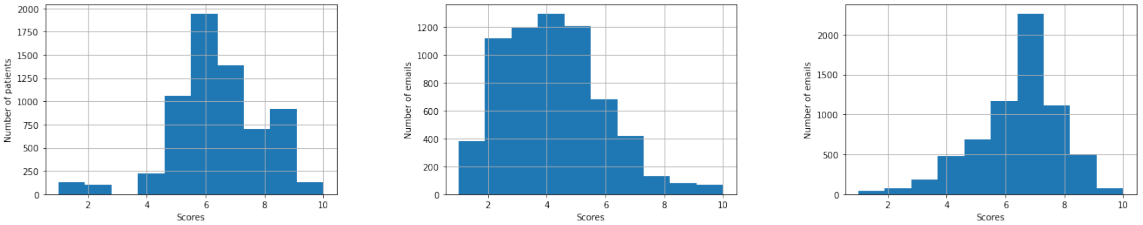
In total, 102 different patients used FITNET, all together sending 6585 emails to their respective therapists. On average, each patient sent roughly 65 emails, each email containing an average of around 120 words. The patient with the least amount of emails only sent 10, whereas the patient with the most amount of emails sent 130. The shortest emails contained only 1 word, often being ‘Gedaan’, which means ‘Done’. The longest email was 2920 words long. The patient who sent the least amount of words in emails only sent 1519, and the patient who sent the most sent 38180 words. On average, each patient sent 7722 words in emails.

2.2.1 Data Selection

While meta-data of the objective data also exists, such as the name of the therapist and when each email was sent, this information was not used. Additionally, the subject lines were not used for this research. This was done because the vast majority of the subject lines have a standard subject, containing the word ‘E-consult’ followed by a number and a description. 4657 out of 6585 had this in their subject line, thus it was decided that it would most likely not add a lot of information. Seeing as the emails were for the patients’ therapy, these subject lines were simply a description of where in their therapy they were. Thus, they, too, were disregarded. The only data used was each patient’s identification number and the bodies of their emails. The identification number for each patient was only used for administration, whereas the bodies of their emails were used to create the model inputs.

2.2.2 Adding Mental Health Scores

To be able to compare each patient’s language use to their mental health scores, these needed to be added to the data. These scores were the same scores from the subjective data, which were explained in Section 2.1.1. These were the scaled STAIC, CDI, and CVSSE scores, as explained in Section 2.1.3. As both the subjective data and the objective data used the same anonymised patient identification numbers, the scores could easily be



(a) Anxiety

(b) Depression

(c) Self-Efficacy

Figure 3: Histograms denoting the number of emails labelled with each value for all three mental health diagnoses for the objective data. On the X-axis, the number denotes the score, and the number on the Y-axis denotes the number of times each value occurs in the data set.

retrieved from the subjective data and added to each email in the objective data. In Figure 3, the distribution of all the output classes over the number of emails can be found. This is similar in shape to Figure 2, but as not each patient sent the same amount of emails, they are not the exact same.

To be able to see if the scores from the subjective data are correlated in any way with the objective data, the STAIC, CDI and, CVSSE scores were also randomly generated for each email. Both data sets were used and the results were compared. Whether or not these predicted results are statistically significant was calculated using an independent T-Test comparing the two sets of predicted values. The results of the models trained on both the labels from the subjective data set and the randomly generated models can be seen in Section 3.2.2.

2.3 Methods to Minimise Impact of Data Set Size

The data set used is relatively small for an AI application to be trained properly. At 102 patients present in both the subjective and objective data set, and a total of 6585 emails in the objective data set, this is significantly smaller than most AI research uses, which can have up to thousands of independent data points. This is mainly because AI models use a large section of the data to train on and try to find commonalities in this section of the data. If the total data set is small, then the amount of data the model can train on is smaller, too. This will influence the maximum achievable performance, also depending on what exactly is in the training data. This is because, if the training data is small, and in it are a few outliers, these will influence the model significantly. Had the training data been larger, the same amount of outliers would not have been as influential in the resulting model.

By training on a too-small data set, or training for too long on any data set, overfitting might occur. Overfitting is when the model has been trained too much and gives great results on the training data, but significantly worse results on the validation and test data. This means that the model is no longer generalisable but only works really well on the data it learnt from. An example of this would be when a model starts remembering data points it has seen and just remembers their answers. Another example would be that the parameters of the model are changed in such a way that even outliers in the training set are classified correctly. As a model that also works well on data outside of the training set is desirable, overfitting is to be prevented.

As was said in Section 2.1.3, the original range of the scores for anxiety, depression, and self-efficacy was not usable with only 102 patients. Thus scaling was applied so that the scores were integers ranging from 1 to 10. These 10 different values were distributed over 102 data points in total. Even if these 10 classes were spread evenly over the data points, that would still result in only 10 patients per class. As the classes are not balanced, this results in classes that have even fewer data points. Having such small classes makes finding commonalities in a class much harder.

This is not to say that it is impossible to train models based on this data, only that there are some methods necessary to prevent overfitting, and even then the resulting models might still be influenced by outliers. One such method to make the results more reliable and less likely to be influenced by outliers is called K-Fold Cross Validation and is described in Section 2.3.1. Other methods, such as re-scaling the original data, have also been implemented and are explained in Section 2.1.3.

2.3.1 K-fold Cross Validation

It is common practice in classification problems such as these, to split the data into three groups: a training set, a validation set, and a test set. A model is trained on the majority of the data, which is the training set. Then, the model is initially tested on the validation set, and hyperparameters can be optimised. Hyperparameters are parameters that influence the functionality of the model. An example of a hyperparameter that was used in this research is for how many iterations the model trains. A hyperparameter that is more specific to this project would be which data categories described in Section 2.1.1 were used as input for the AI model. Lastly, the test set is used to evaluate the optimised AI models objectively. This ensures that the optimisation is not perfectly fitted onto the final results and makes the results more objective.

The way the data is split is 80% training set, 10% validation set and 10 % test set. This results in a training size of 81, a validation set of 10 and a test set being 11 patients. As both the validation and test set are quite small, there is a chance that the split on which the data the model is trained compared to the data on which the model is tested might accidentally be very easy or very hard, thus leading to untrustworthy results. An example of when classifying an easy test set is if the classes in the test set are all similar to the majority class in the training set, thus the model has seen a lot of data similar to the ones it needs to classify. The opposite would result in a harder-to-classify test set, where the minority class is very prevalent in the test set. As the model will not have seen a lot of data similar to it, it makes classifying it correctly harder. The goal is to have a balanced test set, as to make the evaluation as fair as possible.

For small data sets such as these, it is common practice to use K-Fold Cross Validation. This is a technique that, instead of making one single split of the data for the training and validation set, makes K number of splits and trains K number of models. This research uses $k = 10$, as that results in as close an 80-10-10 train-val-test split as possible with K-Fold Cross Validation. The test set is separated from the rest of the data set. The K-Fold cross validation is then done on the training and validation set together.

Here, where $K = 10$, 10 different splits are made. These splits are then used to train 10 different models. The way this is done is that per model, one of the ten parts of the data is used as the validation set, and the other nine parts together are the training set. Which tenth of the data is used as the validation set changes every model, or fold. This is done in such a way that the data in the validation set in a specific fold is not in that fold's training data, but it still gains more reliable results than training only one model. This is because the possibility of the validation set being a statistical anomaly, either positively or negatively affecting the results decreases significantly, as all data is used as validation data in one of the folds. By taking the average performance over all the folds, a general performance can be measured. A visualisation of K-Fold Cross Validation can be found in Figure 4.

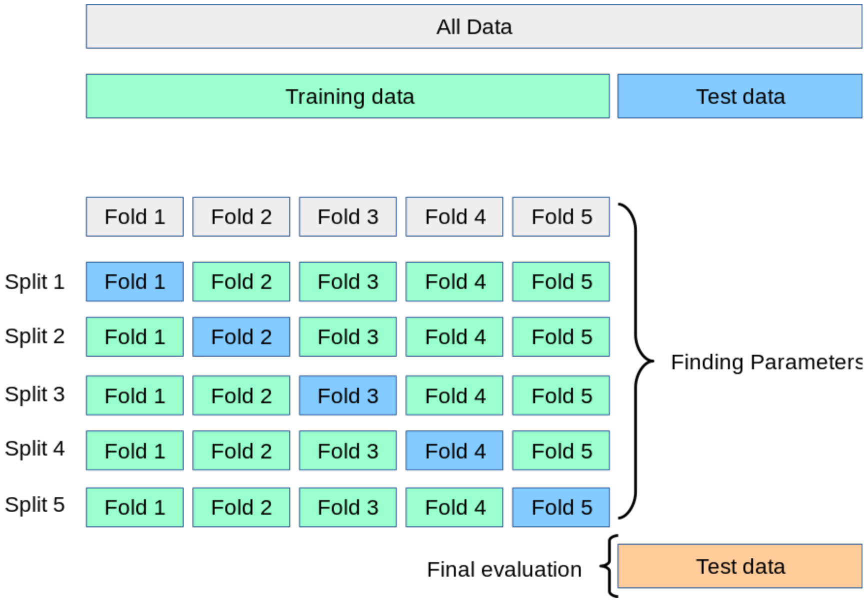


Figure 4: K-Fold Cross Validation example. The data in green is used as the training data in each fold, and the data in blue is the validation data in each fold. After training all five folds, and the optimal hyperparameters are found, the models are used with the test data, which is shown in orange. Source: Scikit-Learn `https://scikit-learn.org/stable/modules/cross_validation.html`

3 Correlation of Anxiety, Depression, and Self-Efficacy Scores with Language Use

This chapter will focus on the correlation between the language used by patients suffering from Chronic Fatigue Syndrome and the scores they received for anxiety, depression, and self-efficacy. The research question this chapter aims to answer is '*Is there a correlation between the language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy and the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires?*'. It is hypothesised that there is a correlation between language use and the level of a patient's anxiety and depression, and the measure of their self-efficacy. While answering this question, a model based on the subjective data will also be trained, with which all subsequent results can be compared.

All the information regarding the method used for creating the models based on the subjective data and the results from these models can be found in Section 3.1. In Section 3.2, all the information pertaining to the method and results based on the objective data is described. The models trained on the objective data are used to answer the research question in Section 3.3, together with other conclusions based on the results for both data types. A visual overview of the models used and discussed in this chapter can be found in Figure 5.

3.1 Approaches to Subjective Data

The approaches described in this section have been done so that it is clear to see how well anxiety, depression, and self-efficacy can be predicted based on the subjective data. Additionally, by knowing how well these models perform, these performance values can be used to compare to the performance values of the models based on the objective data. The

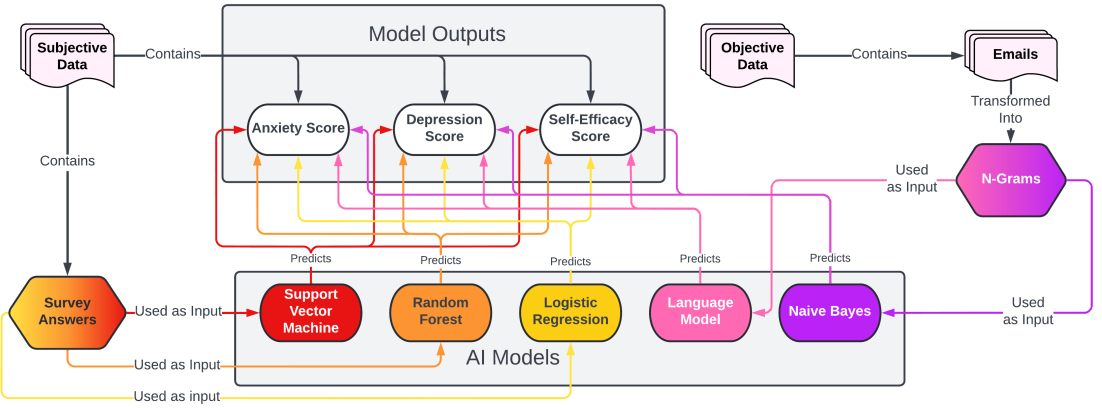


Figure 5: Flowchart showing the general structure of the way the subjective data was used by the models, and what values were predicted.

method for how these models based on the subjective data, as described in Section 2.1, are trained can be found in Section 3.1.1. The results of this method can be found in Section 3.1.2.

3.1.1 Methods Used for Subjective Data Set

Together with the data previously described in Section 2.1, machine learning models can be trained. From the categories described in Section 2.1.1, the categories used are CDC, Symptom Scores, Symptom Meta, and Parents, as this yielded the best results. This chapter describes what models were used, and how they were trained. The three models used are Logistic Regression, explained in Section 3.1.1.1, Random Forest, explained in Section 3.1.1.2, and Support Vector Machines (SVM), which is explained in Section 3.1.1.3. All of these models are created with a similar goal, which is classification. Additionally, they all can only predict one type of class at a time. Therefore, for all three machine learning models, three separate versions were trained, one for anxiety, one for depression, and one for self-efficacy, resulting in a total of nine models. While they have their goal of classification in common, they have different implementations of how to do the classification. For all of these models, K-fold cross validation with K=10, as described in Section 2.3.1, was used.

3.1.1.1 Logistic Regression

Logistic Regression is a statistical model that uses logistic functions to model a dependent variable. As is explained in Section 2.1.4, it requires numerical data to calculate such a model based on logistic functions. Using the input data, a probability for each class is calculated. There are 10 classes after re-scaling (see Section 2.1.3), thus for all 10 classes a probability is calculated, and the class with the highest probability is used as output.

3.1.1.2 Random Forest

Random Forest is a machine learning method that creates a number of decision trees while training. A decision tree is a machine learning method that finds ways to split the data based on its values to predict the correct class. An example of this could be that if all patients with an anxiety score of 6 say that they miss 80% of school, the decision tree would make a split at that 80%. If then data about a patient that only misses 60% of school were to be classified, the decision tree will then not classify the patient as having an anxiety score of 6. A decision tree keeps making splits until there are no uncertainties in the splits left.

Random forest creates many such decision trees, using different splits. Usually, data can be split in many different ways, thus different decision trees are possible for the same data. Not all of these will have the exact same output, as they might split different columns in the data first, thus resulting in different orders. Given the input data, random forest lets all of its decision trees classify the data and then outputs the class that was the output of most of the decision trees through majority vote.

3.1.1.3 Support Vector Machine (SVM)

A Support Vector Machine, or SVM, is another type of classification algorithm. Based on the training data used, it will create a plane or line which separates the data into two different classes. The original SVM only works for binary classification, thus with the data consisting of two classes. As this research has more than two classes, this would not work, thus a multi-class SVM was used. This type of SVM is similar to how random forest utilises multiple decision trees to classify, but a multi-class SVM uses multiple binary SVMs, to then combine those results. For 10 classes, 10 different SVMs are created, where in each SVM one class is put against all other classes. Thus, one of those SVMs calculates the probability of the input being part of one specific class or not. Then, the multi-class SVM takes as output the class where the one-versus-all SVM gave the highest probability.

3.1.2 Subjective Data Results

The results for the three different models trained on the subjective data are shown in Table 4. In the tables in this chapter, two metrics to determine the quality of the output of the model are noted, accuracy and Mean Squared Error (MSE). Accuracy simply states the percentage of a model's output that was correct. For accuracy, random chance of being correct would be 10%, as there are 10 output classes due to scaling the original data. MSE is a way of denoting how far off the model was in its prediction. For a more detailed explanation of MSE, see Appendix B.

It should be noted that while the SVM model performs the best for anxiety prediction, with an accuracy of 31.8%, logistic regression has a lower MSE, at only 3.01. For depression, random forest performs the best for both metrics. Additionally, SVM scores the worst on both metrics, even reaching an accuracy which is lower than chance with 8.2%. The MSE's for self-efficacy are the lowest values for all models, and thus perform the best. Interesting

Table 4: Results for the models using the subjective data. Values in boldface indicate the best scoring model on the metric shown in that column.

	STAIC		**CDI**		**CVSSE**	
	MSE	**Acc**	**MSE**	**Acc**	**MSE**	**Acc**
Logistic Regression	**3.01**	22.7%	6.67	15.5%	1.77	32.7%
Random Forest	4.46	28.1%	**3.95**	**25.5%**	**1.12**	30.9%
SVM	3.87	**31.8%**	8.6	8.2%	2.63	**33.6%**

to note is that the accuracy for random forest is the worst, at 30.9%, this is the model with the best MSE at 1.12. SVM, similarly to anxiety prediction, performs best on accuracy, this time at 33.6%.

3.2 Approach to Objective Data

As opposed to the approaches to the subjective data, the approach to the objective data is based on emails sent by the patients to their therapist. As these emails are not directly about their anxiety, depression, and self-efficacy, the patients are not actively thinking about their symptoms. As it is hypothesised that their language use can be used with an AI model as an indication of their psychological state and well-being, even when not talking directly about their psychological state, these emails are viewed as objective. A more in-depth description of the objective data can be found in Section 2.2.

The goal of the approach to the objective data in this chapter is as follows. It is investigated if the language used by the patients in the objective data is correlated to their anxiety, depression, and self-efficacy scores from the subjective data. In other words, if the scores per patient from the subjective data are more easily predicted by a model than when random labels are attached to the emails. Secondly, the performance of these models will then be compared to the performance of the models trained on the subjective data.

3.2.1 Method Used for the Objective Data

The models used for the approach to the objective data are quite different from the models used in the approach to the subjective data, as the type of data differs greatly between the two approaches. For the objective data, two types of models are used. The first is a probability-based language model, and the second is Naive Bayes, which is also probability based. For probability-based models, the classification is done by calculating the probability of a certain input being part of each possible class. It then checks for which class the probability is the largest, and thus classifies the input as that specific class.

The language model, similarly to the models in the subjective approach, as explained in Section 3.1.1, can only classify one output value at a time. Thus, the same implementation is used three separate times, one for anxiety, one for depression, and one for self-efficacy. As there are 10 classes for each of the three diagnoses, and for each class a single probability model has to be created, a total of 30 models are created every time. As the data is relatively small, K-Fold Cross Validation, as explained in Section 2.3.1, is used, again with $K = 10$. This means that for one run of the program, a total of 300 models are created. The Naive Bayes implementation also used K-Fold Cross Validation with a $K = 10$. For both types of models, the patients who appear in the test set are ensured to not appear in any of the folds.

Two implementations for probability models were made. Both are based on the same principle, which is the word or words used in the emails, also known as n-grams. The n in n-gram denotes the size of the sequence from the sentence used. Unigrams are n-grams of size one, or single words. The model based on unigrams is explained in Section 3.2.1.1. Bigrams are n-grams with n=2, or sequences of two words. Both bigrams and unigrams are used in the implementation of Naive Bayes, which can be found in Section 3.2.1.2. A more in-depth explanation of how bigrams and other higher-order n-grams can be used in language models can be found in Appendix A.

3.2.1.1 Unigrams

As described before, unigrams are single words. In the sentence ‘Anna walks to school’, there are four unigrams, i.e. ‘Anna’, ‘walks’, ‘to’ and ‘school’. To use unigrams as a way to calculate $P(w)$, the probability of a word occurring in a certain text, the formula $P(w) = \frac{Count(w)}{N}$ is used, where $Count(w)$ is the number of times word w was seen in the data set, and N is the total number of words seen. To calculate the probability of a sentence $P(w_1, w_2, ..., w_n)$, the formula becomes $P(w_1, w_2, ..., w_n) = P(w_1) \times P(w_2) \times ... \times P(w_n)$[1].

As for the implementation, this is done by using the training data in each fold to create such a model for each label. Thus, in each model, the probabilities for all the words that were seen for that class are calculated. Then, when given an email as input, the probability for each word in the email is called from the language model and multiplied by the probabilities of all the other words in the input. This is done for all 10 language models, which results in 10 probabilities. The highest probability is then assumed to be the correct class.

This implementation, however, has one downside. If a word occurs in the email that was not seen in the training set, the probability of that word would be $\frac{0}{N}$. This means that that probability would be 0.

Thus, smoothing is required. Smoothing is a way to prevent the probability of unseen words to be zero. This model uses Laplace Smoothing, which means that the count of every unique word gets α added to it. This results in the fact that the total number of

[1] To illustrate this with an example using the sentence ‘Anna walks to school’. Say that ‘Anna’ occurs once, ‘walks’ occurs twice, ‘to’ occurs thrice times and ‘school’ occurs once as well in the whole data set, and that the total number of words seen is ten. Thus, the probability of ‘Anna’ occurring in the text is $\frac{1}{10}$, the probability of ‘walks’ is $\frac{2}{10}$, and $\frac{3}{10}$ and $\frac{1}{10}$ for ‘to’ and ‘school’, respectively. The total probability of the entire sentence occurring thus becomes $\frac{1}{10} \times \frac{2}{10} \times \frac{3}{10} \times \frac{1}{10} = 0.0006$.

unique words also increases with the total number of words multiplied by α. Thus, the new probability of each word becomes $\frac{Count(w)+\alpha}{N+\alpha\times V}$, where V is the total number of unique words. Thus, as in this implementation $\alpha = 1$, the probability for each word is $\frac{Count(w)+1}{N+V}$. Due to Add-K Smoothing, the probability of unseen words becomes $\frac{1}{N+V}$, as also the unseen words get the value of α, in this case 1, added to their count. By calculating the probabilities in such a manner, the language models can also work with unseen words, without the probability becoming zero.

3.2.1.2 Naive Bayes

A simple language model solely based on the probability of a certain n-gram appearing in a class, such as the one described in Section 3.2.1.1, works best if each class contains the same amount of data. In that case, for each class the denominators would be the same, thus the numerators truly influence the resulting probability. This means that if an n-gram occurs more often in one class than it does in another, the probability for that class will be higher.

For example, take two classes A and B, both of which have a total of 20 n-grams. In the data belonging to class A a specific n-gram occurs once, and for the data pertaining to class B the same n-gram occurs twice. The case occurs that a piece of text needs to be classified as belonging to either class A or B, and this piece of text is only that one n-gram. This would mean that the probability of the text belonging to class A is $\frac{1}{20}$, whereas the probability of the n-gram belonging to class B would be $\frac{2}{20} = \frac{1}{10}$. The language model would predict that the piece of text belongs to class B, simply because the probability is twice as high, as it occurred twice as much in the training set.

However, if not all classes contain the same amount of data, this is not always the case. As the denominator is then dependent on the size of the class, n-grams have to occur less often to yield the same probability. This means that if there is a class which contains relatively little data, such as class 10 for STAIC, CDI, and CVSSE, as can be seen in Figure 3, all the probabilities for that class will be significantly higher than those of classes which contain more data. This means that a simple language model will often classify emails in these classes, solely because the respective probabilities of those classes are higher.

To take the previous example with classes A and B, but now class A only consists of a total of 8 n-grams, whereas class B still contains the same 20 n-grams as before. Both data sets still contain the same n-gram the same number of times as before, thus once for A and twice for B. Then, when the same classification task comes, the probability of the text belonging to class B is the same as before, which was $\frac{1}{10}$. The probability for class A, however, has changed to $\frac{1}{8}$, which is larger than the probability of class B, even though it occurred only once in the training data of class A.

A way to work around this is using Naive Bayes, implemented using NLTK (Bird et al., 2009). Naive Bayes can be seen as an extension of a language model, where additional features can be added. One of these features is the size of the data it was trained on. This thus means that the size of each class is taken into account, thus influencing the final classification.

To continue with the example used, some new probabilities would be added to the calculations. As class A contains 8 of the 28 n-grams in the complete training set, all the probabilities while classifying will be multiplied with $\frac{8}{28} = \frac{2}{7}$. For class B, this new multiplier will be $\frac{20}{28} = \frac{5}{7}$. Thus, when classifying the text used previously, the probability for class A would be $\frac{1}{8} \times \frac{2}{7} = \frac{1}{28}$, and for class B this would be $\frac{1}{10} \times \frac{5}{7} = \frac{1}{14}$. Thus, as $\frac{1}{14}$ is larger than $\frac{1}{28}$, the model would classify the input text as class B. This example shows that Naive Bayes thus takes into account discrepancies in the size of the training sets for all the classes.

There are two main differences between the implementation of the language model described in 3.2.1.1 and Naive Bayes. Firstly, the Naive Bayes implementation does not use all the words in the entire data set. It only uses the most n-most popular words, due to computational constraints. This n has been increased in steps by 100 to see the influence on the results. The highest n used was 3000. Additionally, Naive Bayes has been implemented so that it does not only use singular words (unigrams) as features, but it can also use higher order n-grams, in this case bigrams. For this implementation, which uses both unigrams and bigrams, the 3000 most used n-grams were also used as features, once again increased using increments of 100 n-grams.

Secondly, as the Naive Bayes implementation was used as an extension of the language model, it was used as a possible improvement of the results. This means that there were no Naive Bayes models trained with the randomised data.

3.2.2 Objective Data Results

The results based on the objective data are shown in Table 5. For all three output values, the models trained with the labels from the subjective data perform significantly better than the models trained with randomly generated labels. Only the model predicting the depression score performed better than the models based on the subjective data on accuracy with 29.2%, but not on MSE. As for the random model, for all three output values, the model performed around random chance, with the accuracies ranging between 9.8% and 10.4%. Important to note is that the classes that are predicted by the data models are not spread evenly. For all three output values, the model predicts the smallest classes a lot more than other classes.

The results for Naive Bayes are visible in Figure 6. Both the results on the validation set and the test set have been shown, as these are quite different. For the validation set

Table 5: Results for the models using the objective data. The data model used the anxiety, depression, and self-efficacy scores from the subjective data. The random data model used randomly generated values. The differences between the predicted values by the data model and the random model are all statistically significant. Values in boldface perform better than the models based on the subjective data.

	STAIC		**CDI**		**CVSSE**	
	MSE	**Acc**	**MSE**	**Acc**	**MSE**	**Acc**
Data Model	7.24	25.8%	8.95	**29.2%**	7.91	27.1%
Random Data Model	17.6	10.2%	15.8	9.8%	16.4	10.4%

results, both the unigram model and the uni- and bigram models, the shape of the results are quite similar, as can be seen in Figures 6a and 6c. For the validation set results, it can be noted that the depression score is easiest to classify when using Naive Bayes. When comparing both Figures 6a and 6c, it is shown that regardless of whether or not bigrams are used, nor how many n-grams are used, classifying the depression score yields the highest score.

This is different when looking at the test set results, shown in Figures 6b and 6d. Firstly, the general shapes of these are quite different from the shapes of the graphs for the validation set. Secondly, self-efficacy yields better accuracies for both models, as opposed to depression on the validation set.

For the validation set results, the models trained on only unigrams scores higher than the models trained on both uni- and bigrams, even when using the same number of features. For example, at 1000 features, when only using unigrams, CDI is predicted with a 30% accuracy. When both unigrams and bigrams are used, this accuracy drops to around 23%. When looking at the test set results, this difference becomes a lot smaller. While the models based on unigrams still perform somewhat better than those based on unigrams and bigrams, the difference usually is between 2% and 4%.

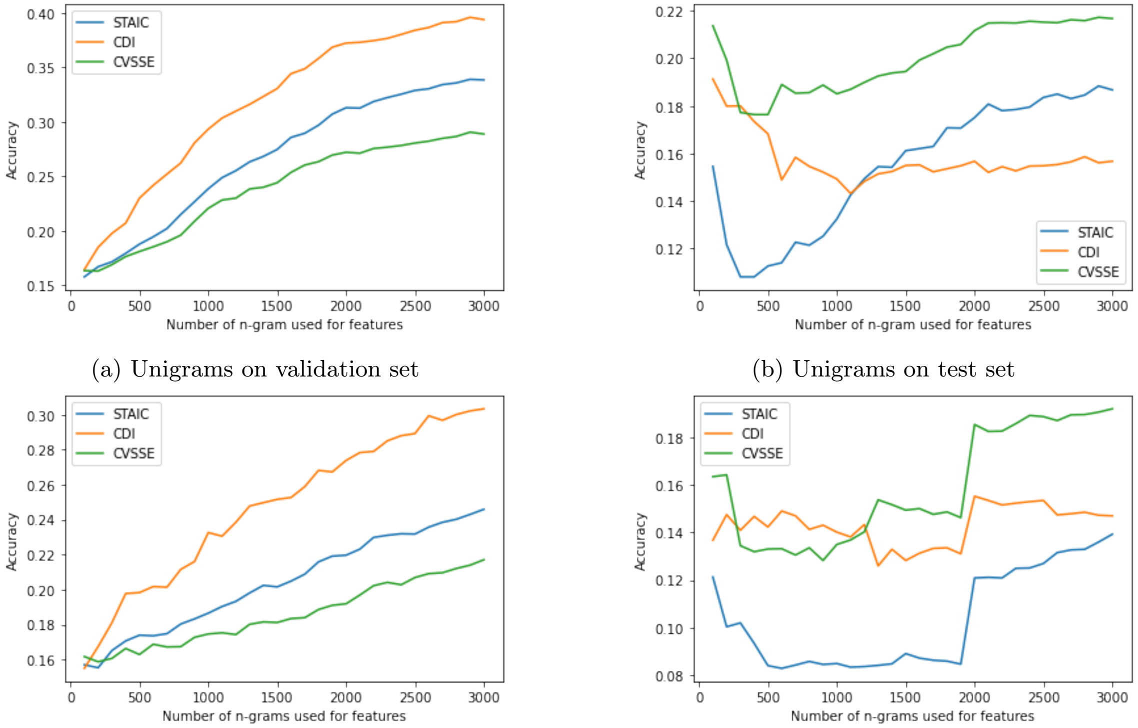
Something else of note is the fact that at 2000 features, the predictions for the uni- and bigram model on the test set have a sharp spike up, for all three output values. Additionally, after 2000 features, most of the models seem to plateau somewhat with the test accuracy. This is not the case for the validation accuracy.

When comparing these results with the results from the unigram language model, Naive Bayes yields lower accuracies than the language model on the test set. They are thus also all lower than the accuracies from the models trained on the subjective data, as described in Section 3.1.2. The accuracies for the validation set yield around the same results for CVSSE as the unigram-based language model. In the case of the STAIC and CDI scores, the accuracies are higher, and also better than the models trained on the subjective data.

3.3 Conclusion

Based on the results discussed above, multiple conclusions can be made. Firstly, and most importantly, the answer is found for the research question '*Is there a correlation between the language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy and the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires?*'. The results have shown that there is a difference between the output of the model that used the labels from the subjective data set and the output of the model using randomly generated labels. This difference, which is statistically significant, implies that there is a correlation between the labels from the subjective data set and the emails in the objective data set, thus confirming the hypothesis.

Secondly, using a unigram model often does not yield results that are better than the models trained on the subjective data. Only the depression score can get predicted more accurately based on the unigram models than it did based solely on the subjective data. The accuracy for depression prediction also yielded higher accuracies than those for



(a) Unigrams on validation set

(b) Unigrams on test set

(c) Unigrams and bigrams on validation set

(d) Unigrams and bigrams on test set

Figure 6: Mean accuracies for models with different numbers of most used n-grams utilised as features.

anxiety and self-efficacy. This might be due to the fact that the distribution of the labels for the depression scores is spread more evenly than they are for anxiety and self-efficacy (see Figure 3). Anxiety and self-efficacy each have one or two significant majority classes in the data, whereas the four largest classes for depression all have a similar distribution. These majority classes are not the ones that are predicted most, as the minority classes get predicted a lot more than they occur.

Thirdly, Naive Bayes has limited capabilities for this classification problem. Seeing as that the validation accuracy kept increasing but the test accuracy seems to plateau, overfitting might also be an issue. When looking at the test set results, it gives worse results than the unigram-based language model. A possible explanation for this is that while Naive Bayes takes the size of each class into account, the implementation used does not use all the words in each email, whereas the unigram-based language model does.

In conclusion, the results support the hypothesis that there is a statistically significant correlation between language use of CFS patients and their scores for anxiety, depression, and self-efficacy. How these scores can be predicted with more advanced models can be found in the next chapter.

4 Predicting Anxiety, Depression, and Self-Efficacy Scores and CFS Recovery From Language Use

In the previous chapter, a correlation is shown between language use and anxiety, depression, and self-efficacy. This chapter will take this information one step further by answering the second research question '*To which extent can language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy be used by AI-models to predict the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires, and whether they will recover from CFS through this therapy?*'. It is hypothesised that all of these features can be predicted to a certain extent. The accuracy of the predictions will likely be limited due to the size of the data set used. This hypothesis will be tested using the same scores for anxiety, depression, and self-efficacy as in the previous chapter, along with the information if the patients recovered from their CFS through the FITNET cognitive behavioural therapy as output variables for multiple AI models. In other words, models were trained to predict the patients' anxiety, depression, and self-efficacy scores and whether or not they recovered from CFS, based on their language use.

As was shown in Section 3.2.2, there was a definite correlation between the language use and the severity of one's anxiety, depression, and self-efficacy. However, using a simple probability-based language model based on n-grams did not yield high accuracies. This chapter aims to find ways to more accurately predict the anxiety, depression, and self-efficacy scores, as defined in Section 3.1, as well as predict whether a patient recovers from their CFS, by using the emails they sent to their therapists.

The recovery data was acquired similarly as the STAIC, CDI, and CVSSE scores, which is explained in Section 2.1.1. The main difference between the recovery values and the other three output variables is that the recovery information is input as categorical data in the data set. To make the value numerical, if the patient was not recovered, they got the value 0, and if they were recovered, they received the value 1. Nijhof et al. (2012) used multiple different measures to determine whether a patient was recovered, including the Columbia Impairment Scale (CIS) (Bird et al., 1993), which is the measure for recovery used in this thesis.

As was described in Section 2.1.4, the machine learning algorithms used require numerical data. However, emails are text, and therefore not numerical. Because of this, three ways to represent text-based emails as numerical vectors have been explored, and all these representations have been used as input for the machine learning models. These three ways are explained in Section 4.1.

The data then was used with two different AI models, Logistic Regression and Neural Networks. An explanation of these two models can be found in 4.2. A visual overview of how the prediction of the different values has been implemented, using BoW, Non-Agentic Language features and the BERTje embeddings with the different AI models used can be found in Figure 7. The results for both AI models are shown in Section 4.3. Lastly, the conclusion of chapter is discussed in Section 4.4.

4.1 Three Ways of Representing the Data Numerically

As the emails written by the patients are not numerical, different ways of representing them as numerical data have been explored. The first, counting how often non-agentic language is used, is made as earlier research has found that non-agentic language is significantly more often used by patients who do not recover from CFS than patients who do (Dalmaijer et al., 2021). The second method is a relatively simple method called Bag of Words. The last method is based on BERTje (de Vries et al., 2019), the Dutch version of BERT (Devlin

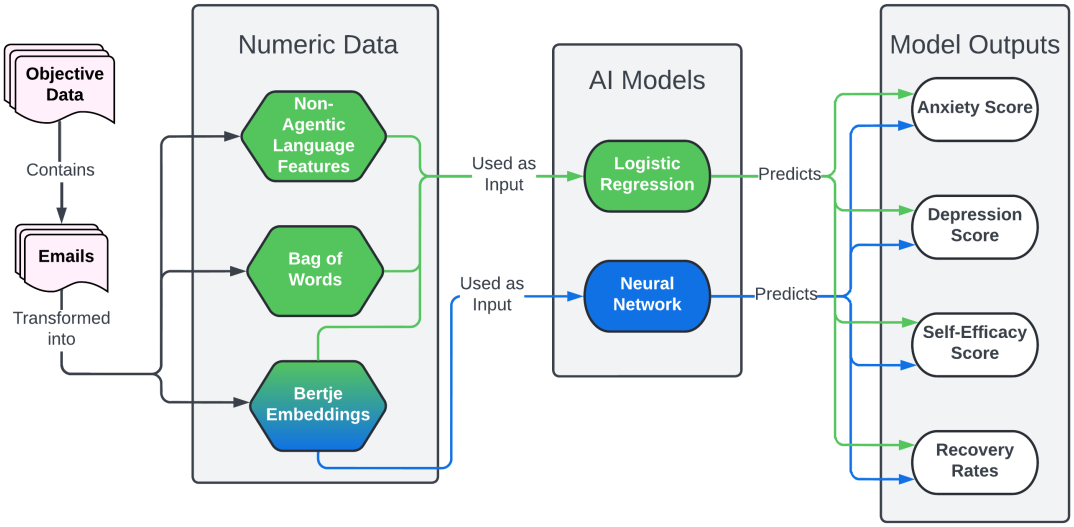


Figure 7: Flowchart showing how the data was used by the models, and what output values were predicted.

et al., 2019). BERT is relatively new but has quickly become an industry standard when dealing with an NLP problem. All of these are described in the following sections.

4.1.1 Counting the Number of Non-Agentic Language Features

Non-Agentic Language (NAL) describes sentence structures where the person writing the text, in this research the patient, is not the actor of the sentence even though they could or should be. These structures can occur in both written and spoken language. An example of NAL would be sentences where passive structures were used. The reason non-agentic language was used as a feature to train the machine learning models in this research, is because patients who recovered from CFS while using FITNET used such sentence structures significantly less than patients who did not recover (Dalmaijer et al., 2021).

In order to be able to count the Non-Agentic Language Features (NALF) in the text, abstractions were made using Part-Of-Speech tags (POS-tags). Part-of-Speech (POS) tagging is a way of making clear what function a word has in a sentence. POS-tags denote nouns as nouns, verbs as verbs, pronouns as pronouns, et cetera (Jurafsky and Martin, 2014). Oftentimes, POS-tags do not only denote the word type, but also additional features about the specific use of the word. For example, if a noun is plural, if a verb is used as a past participle or if a pronoun is second or third person. Which features can be denoted depends on what kind of POS-tagging notation is used. For this research, the POS-tag notations described in van Eynde (2004) were used. These notations are developed for tagging Dutch words and are used by the ALPINO POS-tagger used in this thesis (van Noord, 2006). This POS-tagger is commonly used to acquire POS-tags for large amounts of text written in Dutch and attaches POS-tags to text per sentence.

The NALFs used are as follows, and explained in the subsequent paragraphs: verbal passives, indefinite nouns and pronouns, adjectival passives, nominalizations and, lastly, impersonal constructions. The set of constructions dealt with by Dalmaijer et al. (2021) was a subset of the constructions used in this thesis.

Verbal Passives The Dutch language has a lot of ways that verbs can be used in a passive way. A lot of these are described in Verhagen (1992). The passive structures described by Verhagen (1992) have been counted using both regular expressions and using POS-tags. Any passive structures using a finite verb conjugation of one of a set of auxiliary verbs in combination with a past participle were counted in the program. The auxiliary verbs used were ‘worden’ (to be or to become), ‘raken’ (to become), ‘zijn’ (to be), ‘krijgen’ (to receive). An example of a sentence where a verbal passive is used is ‘Voor mijn gevoel moeten er echt bergen verzet worden voordat ik weer gezond ben’ (lit: For my feeling mountains truly have to be moved before I’m healthy again).

Indefinite Nouns and Pronouns Sentences that utilise indefinite nouns and pronouns do have an active structure, but the subject of the sentence is not the same as the actor, which in this case would be the patient. For the scope of this thesis, this structure keeps track of when the patient uses second-person singular pronouns and third-person singular and plural pronouns. For the second person singular, this was done both in subject, object and possessive form. The forms used are 'je', 'jij', 'jou' (all meaning you) and 'jouw' (yours). An example sentence using the second person pronoun in such a matter would be 'Je bent je er op het moment dat je daaraan denkt niet zo van bewust' (lit: You are in the moment that you think about that not that aware of it). This sentence could have been written in an active way, where the patient used the first person pronoun, but, albeit subconsciously, they removed themselves from the sentence.

For the third person pronouns, this was only tracked if they were the subject, both for plural and singular. It was also made sure that all the pronouns in the subject space were congruent with the verb. In other words, if the subject was singular but the main verb was not, it was not counted. An example of a sentence where a third-person pronoun would be indefinite is as follows: 'Daardoor kunnen ze het moe zijn verminderen.' (lit: Through that they can reduce the being tired.) The 'ze' (they) in this sentence refers to the therapists or the doctors, but not to the patient themselves, even though they are the ones that will have to do the most work to get better.

Lastly, the word 'men' was counted. Men is a word in Dutch that translates roughly to 'one' or 'people'. An example sentence using men would be 'Men moet hard werken om beter te worden.' (lit:'One has to work hard to get better'). While this sentence is correct, the patient clearly removes themselves from the action. They have phrased it as a factual statement, not something they themselves do.

Adjectival Passives Oftentimes through the usage of a suffix, a verb is transformed into an adjective, thus creating an adjectival passive. This, too, removed the actor, because it allows for a verb to be used in an adjectival way, as opposed to requiring a subject. An example sentence would be: 'Er zijn niet echt afspraken gemaakt over de gemiste lessen.' (lit: There have not really been any agreements about the missed classes). Here, 'gemiste' comes from the past participle 'gemist' of the verb 'missen' (to miss), and is used in an adjectival way. The way the adjectival passives were found was by using the POS-tags, as they are one of the characteristics that ALPINO uses is the 'prenom', which indicates verbs used as an adjective.

Nominalizations Where the adjectival passive is an adjective created from a verb, a nominalization is where a noun is created from a verb. For example, how 'verbetering' (improvement) is created from the verb 'verbeteren' (to improve). Nominalizations such as these also allow a sentence to be phrased so that the speaker is not clearly the agent. An example would be 'En wat te doen als het niet helpt en ik geen verbetering zie?' (lit: And what to do if it does not help and I do not see improvement). By phrasing it in this way, the improvement is not clearly an action undertaken by the patient, but almost becomes something that just happens *to* the patient. The main basis for the counting of nominalisations came from their structure, as described in Coppen et al. (2012). From these different structures, multiple ways of counting were created. If a noun uses the article 'de' or 'een' and if it ends in '-ing' or 'atie', then it was counted as a nominalization. For the other forms of nominalizations, the data present in the POS-tags acquired using ALPINO was used (van Noord, 2006). If the word's article was 'het' or 'een' and the word itself was an infinitive form of a verb, it was counted as well. Additionally, if a word was a verb according to its POS-tag, and had the characteristic 'nom' in the POS-tag, it was counted as well.

Impersonal Constructions There were two different types of impersonal constructions that were counted. The first was a specific usage of the Dutch word ‘er’ (there). The word er can be used in multiple contexts and ways, but only one denotes the usage of non-agentic language. Therefore, this is the only usage that will be explained here. This usage of er is known as a ‘plaatsonderwerp’ (topical subject) (Grondelaers et al., 2007). In other words, it takes the place of the subject of a sentence, where the patient could have used a first-person singular pronoun. An example is: ‘Er werd niet veel gedaan.’ (lit: There was not a lot done). Er was counted if it was used in combination with a past participle and a form of the auxiliary verb ‘worden’ (to be or to become).

The second type of impersonal constructions used were infinitivals. This is where the main verb is used as an infinitive, possibly in combination with other words also denoting infinitive usage. The way this was counted was by keeping track of the usage of the words ‘om’, ‘te’ and the verb ‘gaan’, and to see if a separate infinitive was used, which was checked using the POS-tags. ‘Om’ and ‘te’ are used in Dutch in a way similar to the English ‘to’ in ‘to be’ when talking about infinitives.

Distribution of NALFs over Output Variables In Figure 8, the differences are shown in frequencies of non-agentic language between patients who did and did not recover. It is important to note that these frequencies are not in line with the frequencies shown in Dalmaijer et al. (2021). They found that patients who did not recover from CFS used significantly less non-agentic language than patients who did recover, whereas here the difference in NALF use is very similar for both groups. Dalmaijer et al. (2021) did not compare non-agentic language use and anxiety, depression, and self-efficacy, only whether the patient recovered or not. The frequencies for the non-agentic language use and these scores can be found in Figure 9, and as Dalmaijer et al. (2021) only looked at the relationship between NALFs and recovery, and not anxiety, depression, and self-efficacy, can thus not be compared to their results.

Important to note is that as the NALFs were abstracted, it is possible that some instances of features were missed, or that some instances were counted that should not have been. An example of instances being counted when it should not have been, can be shown based on how indefinite pronouns were counted. As it was counted how often a ‘zij’ or ‘ze’ occurs in combination with a verb in the plural position, the idea was that this would be about the doctors or therapists. In practice, this could also be about other people, such as in the sentence ‘Zij gaan dan wel naar school’ (lit: They are then going to school), where the ‘zij’ would refer to classmates. As by simply counting instances of NAL the meaning of previous sentences is not taken into account, this could not have been prevented.

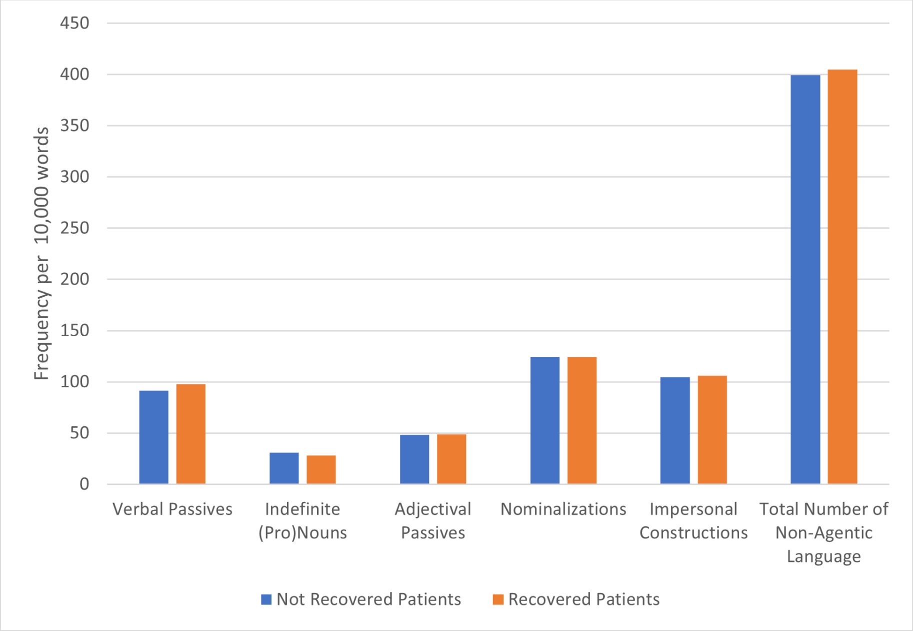
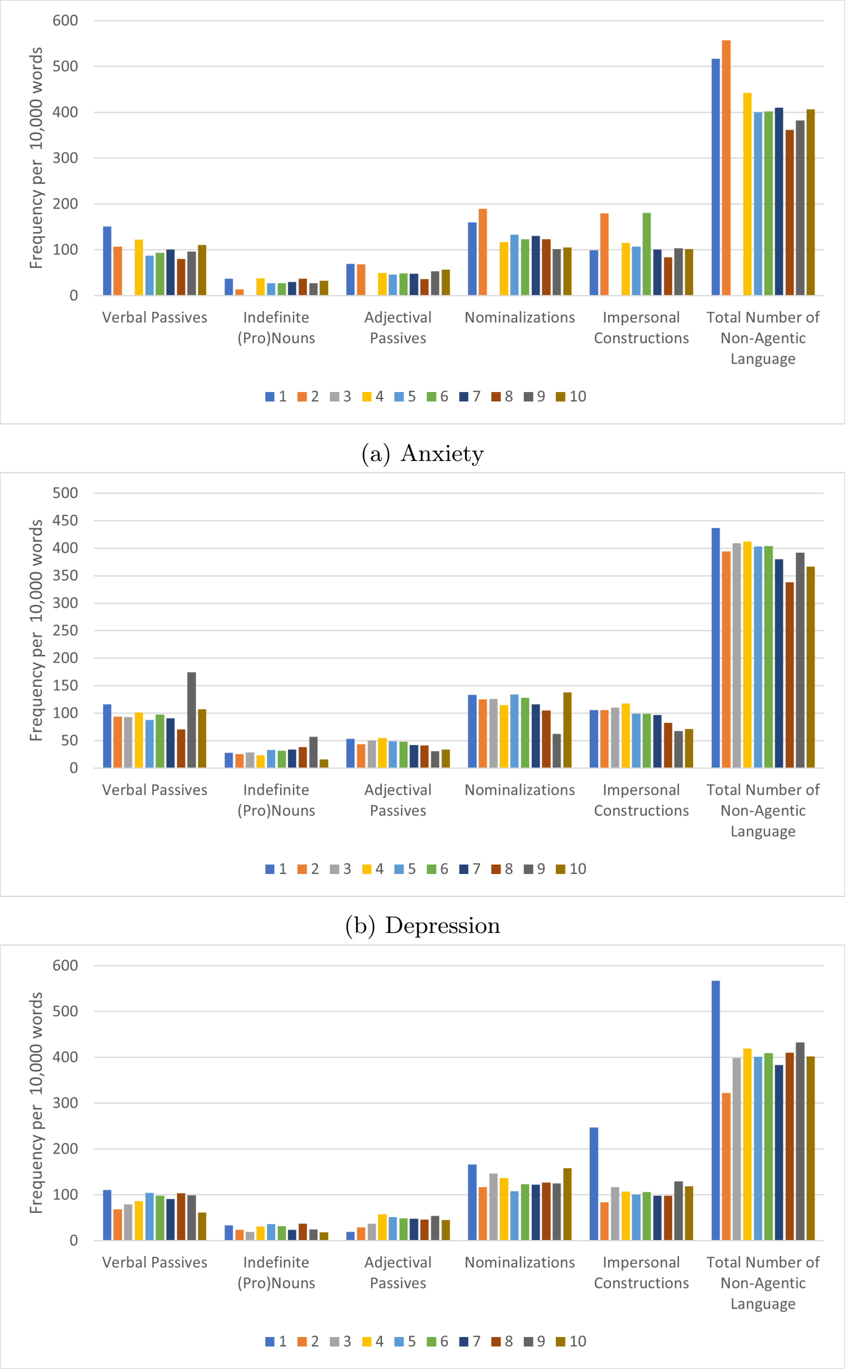


Figure 8: Frequencies of Non-Agentic Language Features used by both recovered and not-recovered patients, which are almost the same.



(a) Anxiety

(b) Depression

(c) Self-efficacy

Figure 9: Frequencies of Non-Agentic Language Features for all the 10 scaled labels of the anxiety, depression, and self-efficacy score. Scaling is described in Section 2.1.3.

4.1.2 Bag of Words

Bag of Words (BoW) is a relatively simple data representation, shaped as one big $m \times n$ matrix, where m is the number of unique emails and n is all the unique words in the total data set. The values in each row represent how often each word occurs in one email. BoW is therefore closely related to unigram models (described in Section 3.2.1.1). The main difference between BoW and unigram models is that for BoW the values in the matrix are not used to calculate probabilities, but are static numeric values solely indicating the number of times each word appears in an email. Through using BoW, these counts can be used for different AI models, such as logistic regression. As the way unigrams were utilised previously was probability based, using them as features in logistic regression was not feasible.

To make visualisation easier, an example where the whole data set were to consist of only the following two sentences, 'I have to go to school but I do not want to.' and 'I am not going to school, but on vacation'. There are 13 unique words, so $n = 13$, and 2 sentences, so $m = 2$. Thus, the size of the matrix will be 2×13. The matrix is shown in Table 6.

Table 6: Example of Bag of Words matrix, based on a data set consisting of two sentences. These sentences are 'I have to go to school, but I do not want to.' and 'I am not going to school, but on vacation.' The words used and the different sentences are named in this example, but not in the actual BoW matrix.

	I	**have**	**to**	**go**	**school**	**but**	**do**	**not**	**want**	**am**	**going**	**on**	**vacation**
Sentence 1	2	1	3	1	1	1	1	1	1	0	0	0	0
Sentence 2	1	0	1	0	1	1	0	1	0	1	1	1	1

An extension to Bag of Words was made to make the resulting matrix slightly smaller without losing the words with the most meaning. This was done by removing certain stop words from the Bag of Words matrix. This is common practice in AI applications because stop words usually do not provide a lot of additional information to the contents of texts.

The stop words that were removed in this extension are as follows: ‘het’ (the), ‘de’ (the), ‘een’ (a/an), ‘den’ (the), ‘der’ (the), ‘des’ (the), ‘en’ (and), ‘dat’ (that), ‘dit’ (this), ‘die’ (that), deze’ (these), ‘aan’ (on), ‘naar’ (to, locative), ‘om’ (in order to), ‘onder’ (under), ‘op’ (on), ‘over’ (over), ‘uit’ (out), ‘door’ (through), ‘tegen’ (against), ‘hierin’ (in this), ‘vanaf’ (from), ‘voor’ (for), ‘na’ (after), ‘nu’ (now), ‘ons’ (us), ‘haar’ (her), ‘onze’ (our), ‘hun’ (their), ‘ook’ (too), ‘te’ (to), ‘ten’ (at), ‘ter’ (to), ‘tot’ (until), ‘enige’ (some, only), ‘enkele’ (some), ‘enz’ (etc), ‘etc’ (etc), ‘hoe’ (how), ‘wat’ (what), ‘wie’ (who), ‘is’ (is), ‘zijn‘ (to be) (Vrije Universiteit Brussel, n.d.).

One downside of BoW is that the resulting matrix is quite sparse. This means that a lot of values in the matrix consist simply of zeros. As the number of columns in the matrix is the same as the total number of unique words, and each row represents all the words in an email, most possible words will not be used in most emails. This can also be seen in the example in Table 6. Even though only two sentences are used, almost a third of the values are 0. This is made even worse by spelling mistakes. Due to this, some words get multiple columns in BoW, where one could have sufficed. For example, the Dutch word ‘chagrijnig’, which means ‘grumpy’, was spelled correctly in 16 emails. A common misspelling of the word, ‘chagerijnig’, was present in 15 emails. Other misspellings were also found, such as ‘saggarijnig’ and ‘sagarijnig’. Interesting to note that while a lot of these misspellings were labelled correctly as adjectives by the ALPINO POS-tagger, some of them were labelled as nouns.

Another way to visualise the sparsity of the data is to realise that the total number of unique words in the entire data set is 14711, whereas the average email contains only 120 words in total, and the longest email is 2920 words long. Thus, even when this longest email is converted to a Bag of Words representation, a minimum of 11791 columns would contain a zero, depending on the number of unique words in the email.

Additionally, Bag of Words only takes into account which words are used, but not their order. As language is contextual, it is more than just a collection of words. Grammar and sentence structure add a lot to its meaning (Pennebaker et al., 2003). Even the simple example of the word ‘not’ can drastically change the meaning based on where in a sentence it is placed. The sentences ‘I am not going to school, but to work’ and ‘I am going to school, but not to work’ have very different meanings, but would get the same BoW representation, as both sentences consist of the exact same collection of words. Therefore, BoW is only a simplification of the original emails. Additionally, it does not take into account the meaning of any of the words, simply how often the words occur. Because of this, BoW will also count homonyms, words with the same spelling but a different meaning, as the same word, while their meaning differs.

However, while BoW has these downsides, it was still useful for this research. Through its simplicity, BoW is relatively simple to both implement and understand, thus allowing for the the model to stay explainable. As this thesis uses medical data, and one of its main goals is to make diagnosis easier, it is very important to have a model be explainable. As the healthcare providers need to be able to explain to patients why certain steps in their treatment plan will be taken, the healthcare providers also need to understand the models, and be able to explain these. For this reason, BoW is used in this thesis.

4.1.3 BERTje

The problem of not taking into account word meaning and structure in a sentence is attempted to be solved by BERTje (de Vries et al., 2019), which also decreases explainability. BERTje is an extension of the English model BERT (Devlin et al., 2019), in which the English language information has been replaced by Dutch language information. BERT, and similarly BERTje, are transformer-based models that give vector representations of sentences. These models are trained on very large amounts of data, as to give an as accurate representation as possible. At this point, using BERT in natural language-based AI research is common practice, as there are very few models that yield similar results, and BERT is relatively simple to implement.

The key feature that makes BERT and BERTje especially useful for this research is that they take context into account. Both models do not only look at the words themselves but also at where they are used in a sentence, at what words they are combined with, etc. This also means that when looking at homonyms the vector representing each word will differ. For example, the BERT vector representation for a bank that contains money will have different values than the vector representation for a bank of a river.

BERTje has a maximum input length of 512 tokens, in this case words. This means that for all emails sent by the patients, only the first 512 words were used to get a vector representation of the email. The subsequent words were ignored. Additionally, the longer the input, the larger the vector, and the longer the computational time. Therefore, models were also trained with a maximum input length of 256 tokens. For the models using the maximum length of 256 tokens, 808 emails of a total of 6585 were shortened. For the models that utilised vector representations based on a maximum of 512 tokens, 227 out of 6585 emails were shortened.

BERTje was used for Logistic Regression by calculating the vector representations for each word in an email, and then taking the average of all of the second to last hidden layers of each word[2]. Hidden layers are part of the model that BERT and BERTje use to calculate the vector representation. For more information on how the Logistic Regression was implemented, see Section 4.2.1.

[2] Implementation inspired by `https://mccormickml.com/2019/05/14/BERT-word-embeddings-tutorial/#why-bert-embeddings`. The author of this blog post calculates a vector per sentence, but in this research a vector was calculated per email.

For the neural network, the BERTje vectors were used slightly differently[3]. This is because the neural network that was used uses the hidden layers directly. Logistic regression does not use hidden layers, but neural networks do, therefore there was no need to combine these layers for the neural network.

4.2 Two Models Used to Train on Objective Data

After creating the multiple numerical representations of the emails, these values were used to train multiple AI models. Two classification models were used, both in different ways. These two models, Logistic Regression and Neural Networks, predicted all four values. Logistic Regression is explained in Section 4.2.1, and Neural Networks are described in Section 4.2.2. Logistic Regression utilised K-Fold Cross Validation with K=10 (see Section 2.3.1). However, the Neural Networks did not use this due to computational time constraints, as training one neural network took between 20 and 24 hours. The neural network used a static training, validation, and test set that was defined beforehand.

All results of the models are compared to the results from the models trained on the subjective data set, as described in Section 3.1. The values used to compare performance on prediction of anxiety, depression, and self-efficacy scores are those from Table 1. For this comparison, the best values of the best-performing models are used. As these models did not predict the patient's CFS recovery, this cannot be compared. The performance of the recovery prediction models will be compared to an accuracy of 61.6%, as this is the percentage of emails sent by a patient that was recovered in the entire data set. In other words, if a model were to always predict that a patient recovered, regardless of input, the accuracy would be around 61.6%.

[3] Implementation used by `https://lajavaness.medium.com/regression-with-text-input-using-bert-and-transformers-71c155034b13` The author of this blog post uses CamenBERT, the French version of BERT.

4.2.1 Logistic Regression

Another implementation used for the classification method used is Logistic Regression. The concept of logistic regression is already explained in Section 3.1.1.1. However, a key difference between the implementation described there and the one here is that this Logistic Regression is based on the numerical representation of the emails.

One key advantage of logistic regression is that which features are used can be easily changed. As all the features described in the previous sections are represented in the code as matrices, each of these can be easily extended using one of the other representations. This means that some logistic regression models were trained using only Bag of Words, or only the non-agentic language features, or only the BERTje vector embeddings, and also others with combinations of these. As was described in Section 4.1.2, an extension of BoW was used where stop-words were removed. This extension has also been used in combination with the non-agentic language features. It is important to note that while some of the stop words that were removed for BoW are necessary to calculate the non-agentic language features, these two processes are done completely separate from each other. This means that the values of the non-agentic language features were not different when combined with the BoW version where the stop words were removed.

4.2.2 Neural Networks

Neural Networks (NNs) work differently than the previous models. NNs are modelled after the human brain, with nodes representing the neurons. These nodes are connected by mathematical calculations, and while training a neural network, the model tries to find the most optimal sequence of calculations.

To find this optimal sequence, a way of comparing the output of the model to the value it was supposed to be has to be defined. This formula is called a loss-function, and can be changed depending on the requirements for each task. Two different types of loss functions were used in this research. For the model predicting the recovery rates, which only has two classes, accuracy was used as a loss function. Thus, it simply tried to maximise how often it was correct.

For the case of STAIC, CDI, and CVSSE, a different loss function was used. If it had used accuracy, and a model was to predict the label 2 for a certain email, but the correct label was 8, it would be viewed as just as wrong as when 7 was predicted when the correct label was 8. As STAIC, CDI, and CVSSE have values on a scale, this is not the case in actuality. The model would be better if, even if it was incorrect, would be close to the true output than if it was far off. Because of this, the loss function used was Mean Squared Error (MSE). For more information on MSE, see Appendix B. The neural network tries to minimise the MSE, in order to get as close to the correct output as possible.

The fact that MSE uses the squared error has the result that the farther the model's prediction is from the actual value, the harsher it gets 'punished'. If Mean Absolute Error (MAE) was used, then if the loss function would see the difference between being 2 off versus being 4 off in the prediction as a twice as wrong result. By using MSE, the loss function makes the difference between this 2 and 4 of the predictions into the difference between $2^2 = 4$ and $4^2 = 16$ in the loss function.

Another key thing about NNs is that they train for a certain number of epochs. One epoch means that the model has seen the entire training set once. The more times an NN has seen the entire training set, the more optimised the model can become, and the lower the values of the loss function will become. It is important to note, however, that this means that the values of the loss function are low for the training set. When ensuring this, while disregarding the results based on the validation set, this can lead to overfitting.

The main way to prevent overfitting with a neural network is to not train a model for too long. This is often done by comparing the values of the loss function based on the training set with those of the validation set. Once the model loses significant performance on the validation set, the training is stopped.

This is not done in this research. This is simply because training the models on the BERTje embeddings was time and computationally heavy, thus they were not trained for a lot of epochs. For the embeddings based on 256 tokens, the model was trained for 20 epochs. For the models based on embeddings from 512 tokens, this was 9 epochs. Both of these took around 20 to 24 hours. During this time, the loss function values of the validation set did not increase compared to those of the training set, thus no overfitting occurred.

4.3 Results

As described in Section 4.2, two different types of models were used. For Logistic Regression, the accuracy scores are shown in Table 7, and the MSE scores for the same models are shown in Table 8. These tables show the results of all the different combinations of numeric representations of the emails that were used. As the accuracy scores between the validation sets and the test sets differ greatly, both have been shown. An example of such a large difference is how the model using only BoW when predicting the CDI score, the validation set has an average accuracy of 43.3%, whereas on the test set this accuracy drops to 15.6%. On the validation set this accuracy is a lot higher than random chance, which with 10 classes is at 10%, but for the test set this is relatively close to chance.

Interesting to note is the fact that for both CDI and CVSSE, on the test set, the accuracies increase slightly when using BERTje and the NALF as compared to when BoW and NALF are used. For CDI this is a difference between 14.8% and 19.5%, and for CVSSE this is a difference between 33.4% and 35.3%. For CVSSE, using the BERTje embeddings

Table 7: Accuracy scores for the logistic regression models. For all three output variables, both the values of the validation set and the test set are shown. The values in red denote where overfitting might play a role in the higher accuracies. Values in bold are the best-performing models per column. Underlined values are values that perform better than the models based on the subjective data, as described in Table 1.

	STAIC		**CDI**		**CVSSE**		**Recovery**	
	Val	**Test**	**Val**	**Test**	**Val**	**Test**	**Val**	**Test**
NAL Only	30.2%	**23.9%**	20.4%	17.8%	31.6%	55.1%	58.1%	87.2%
BoW Only	45.1%	18.3%	43.3%	15.6%	45.7%	33.5%	68.7%	**59.1%**
BoW + NAL	**45.5%**	19.4%	43.1%	14.8%	45.5%	33.4%	**68.8%**	58.4%
BoW excl. Stop words + NAL	**45.5%**	18.1%	**43.4%**	15.8%	**45.8%**	32.8%	**68.8%**	58.1%
BERTje	34.6%	17.7%	31.8%	**19.7%**	34.7%	34.7%	62.7%	53.2%
BERTje + NAL	34.5%	17.3%	31.3%	19.5%	34.9%	**35.3%**	62.7%	52.9%

Table 8: MSE scores for the logistic regression models. For all three output variables, both the values of the validation set and the test set are shown. The values in red denote where overfitting might play a role in the higher accuracies. Values in bold are the best-performing models per column. Underlined values are values that perform better than the models based on the subjective data, as described in Table 1.

	STAIC		**CDI**		**CVSSE**	
	Val	**Test**	**Val**	**Test**	**Val**	**Test**
NAL Only	**3.21**	**3.61**	**3.95**	**3.35**	1.02	1.12
BoW Only	3.22	5.43	4.09	4.93	**3.09**	2.35
BoW + NAL	3.23	5.58	4.11	5.00	3.10	2.35
BoW excl. Stop words + NAL	3.25	5.73	4.05	4.82	3.11	2.43
BERTje	4.15	6.11	4.91	5.52	3.58	2.22
BERTje + NAL	4.14	**6.27**	4.94	5.71	3.61	**2.21**

Table 9: Results of the neural networks trained. For STAIC, CDI, and CVSSE both accuracy and MSE are included. As the recovery is a binary classification, MSE has no additional information. Underlined values are values that perform better than the models based on the subjective data, as described in Table 1.

	STAIC		**CDI**		**CVSSE**		**Recovery**
	MSE	**Acc**	**MSE**	**Acc**	**MSE**	**Acc**	**Acc**
Token size 256	2.13	40.8%	2.74	38.5%	2.11	40.5%	73.1%
Token size 512	2.18	36.3%	2.88	35.6%	2.07	37.9%	71.6%

even allows the model to perform better on the test set than the model based on the subjective data did on the same test set.

Additionally, CVSSE yields better results for all the models, compared to CDI and STAIC. STAIC yields results on the test set between 17.3% and 23.9%, and CDI is around the same range between 14.8% and 19.7%. CVSSE, however, has a range between 32.8% and 55.1%. For this last one, it should be noted that this model predicted the same label, which was 7, 95.8% of the time for the test set. The actual times it was indeed labelled 7 was 56.5% of the time. Seeing as this last value is very similar to the accuracy of the model, this result is put in red.

As for the recovery predictions, it is important to note that the models based on NAL only predict that the patient recovers. For this reason, these values in Tables 7 and 8 are shown in red, as these accuracies are exactly the same as the percentage of patients in the respective data sets that have recovered. As for the other models, BoW performed the best, but all models did not perform much higher than chance. Random chance for this classification is at 50%, whereas the models performed between 52.9 and 59.1%. While they do perform slightly better than random chance, none of the results from the test set are better than 61.6%, which is the percentage of the data set which recovered.

Table 9 shows the results for the Neural Networks. All of the models shown in this table perform better than the models based on the subjective data. Only the CVSSE models perform worse than the subjective data models for the MSE. The MSE for the subjective data model when predicting CVSSE was low, at only 1.12.

As was said in Section 4.2.2, the models using token size 256 trained for a total of 20 epochs, whereas the models using token size 512 trained for only 9 epochs. As can be seen in Table 9, the models using the smaller token sizes perform slightly better. For STAIC, CDI, and CVSSE, the accuracies are between 38.5% and 40.8%. As these classes contain 10 different possibilities, random chance would be 10%, so these values are a lot higher. Also good to note is that the MSE for STAIC and CVSSE is a little higher than 2, which means that oftentimes the model is not that far off of the correct label.

As for the recovery prediction for the model using a token size of 256, this gives an accuracy of 73.1%. This, too, is a lot higher than random chance, which for this class would be 50%. The accuracy score of 73.1% is also higher than 61.6%, the percentage of emails sent by recovered patients.

As for the models using the token size of 512, the accuracies for STAIC, CDI, and CVSSE are a little lower than the ones for the token size of 256, as they are between 35.6% and 37.9%. These, too, are a lot higher than random chance and the accuracies from the models based on the subjective data. As for the MSE, the ones for STAIC and CDI are slightly higher than their 256 counterparts, but the one for CVSSE is slightly lower, even though the accuracy is performing less well. The recovery model, with an accuracy of 71.6% also performs slightly worse than the model based on fewer tokens.

4.4 Conclusion

From these results, multiple things can be concluded, including an answer to the research question '*To which extent can the language used by teenage chronic fatigue syndrome patients be used to predict the severity of their anxiety, depression, or self-efficacy, and whether* *they recover from CFS using correspondence-based cognitive behaviour therapy?'*. The results show that when using the BERTje embeddings combined with a neural network can definitely work to predict fairly decently whether or not a patient will recover. At 73.1% accuracy when using 256 tokens, this is not perfect but does give a very good indication. This confirms the hypothesis.

For 512 tokens, the accuracy of the neural network is 71.6%, thus also very promising. This difference in accuracy, even though the second one uses more data, is likely due to the difference in the number of epochs. While both models have run for a little less than 24 hours, the smaller model had 20 epochs, whereas the larger one only had 9. This means that the smaller model saw the entire training data more than twice as much, and thus had more information to base the optimisation on.

Furthermore, the logistic regression model that predicts the recovery rate based solely on the non-agentic language features severely overfits. As this model only predicts that the patient will recover, seemingly regardless of the content of the email, this result cannot really be taken into account. This is most likely due to the fact that the non-agentic language features are distributed fairly evenly between the recovered and non-recovered patients, as is shown in Figure 8. This is not in line with the conclusion from Dalmaijer et al. (2021), as they found a significant difference between the non-agentic language usage of patients who did recover and those who did not.

One big difference in how the non-agentic language features were counted in this research compared to the method Dalmaijer et al. (2021) used, is that they counted the features by hand, whereas this did not. To be able to let the computer count the non-agentic language features, the POS-tags given by ALPINO were used. It is assumed that all the words in the data set were tagged correctly, but it is not feasible to manually check this. Therefore, it is possible that the tagger made mistakes, which could have affected the completeness and correctness of the counting of the non-agentic language features. This means that further research is required to be able to say if non-agentic language features are a good marker for CFS recovery rates.

Additionally, the self-efficacy model based on non-agentic language features also seems to be overfitting. In this case, the overfitting is not quite as severe as it is for the recovery model, as it does sometimes predict other values, but 95.8% of the time it does predict the majority class, resulting in the high accuracy. A possible explanation for this might be that there are not a lot of features to train on when only using non-agentic language. This could result in the models not finding good ways to optimise the parameters and thus consistently predicting the majority class. For the self-efficacy, the spread of the features, visible in Figure 9c, is less uniform over all ten classes than it was in the case of the recovery, but still might be too similar for the logistic regression model.

Moreover, using BERTje embeddings for logistic regression slightly improves the results for both the depression and self-efficacy score prediction. Unfortunately, it is not clear which features exactly BERTje takes as most important. However, it could be theorised that it also takes into account some information about sentence structures that might be indicative of a patient's level of depression or self-efficacy.

Additionally, predicting the level of a patient's self-efficacy goes a lot better than predicting the state of their anxiety and depression, using logistic regression. Even when disregarding the previously mentioned model based solely on non-agentic language features, in all the other cases, the models for self-efficacy outperform those for depression and anxiety by a large margin. This might indicate that language use is more related to a patient's self-efficacy than it is to their anxiety or depression.

Furthermore, using a neural network to predict any of the values based on language use, whether that be anxiety, depression, self-efficacy or whether the patient recovered, yields better results than logistic regression does. For anxiety, depression, and self-efficacy, these results might work as an indication of how a patient is doing in each of these regards, but as the predictions are all lower than 50%, they definitely cannot be used in actual medical practices. Using a model that is incorrect more than half of the time is not ethical.

Lastly, the results of the neural network most likely can improve with more training time, as the training loss and test loss were close. In other words, the model was not yet overfitting, and could thus still improve. However, this was computationally not possible within the scope of this project. Additionally, more data could also yield better and more reliable results. Both in the sense that it is surer that the test set is more representative of the data, and that the models are less sensitive to outliers, and thus more generalisable.

In conclusion, predicting the recovery rate using neural networks works very well. Using neural networks to predict anxiety, depression, and self-efficacy works better than the models based on the subjective data. As for logistic regression, self-efficacy can be predicted using BERTje embeddings to a point better than the models based on the subjective data. Anxiety, depression, and recovery do not improve compared to the subjective data models when using logistic regression. This is in line with the hypothesis as described in Section 1.4.

4.4.1 Limitations

There are two main limitations. Firstly, as the neural network uses exactly 10% of the data as a test set, this is done differently than it was done for the other models described. In this case, it was not ensured that emails that were written by patients that the model was trained on were not present in the test set. In other words, there is a chance that the neural network has already seen text written by the same patients during training time as well as when testing the model. This could influence the results to be better than they would be otherwise.

Secondly, as the data set is quite small, there is a chance that the test set used by all the other models is not representative of the data. As for that test set, it is 10% of the number of patients, there is a chance that the patients in that 10% are not representative, or may have written fewer emails than other patients. This would be a possible explanation for why, for the logistic regression models, there is such a large discrepancy between the accuracies for the validation and the test set.

5 Discussion

This study focused on two main research questions. The first was '*Is there a correlation between the language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy and the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires?*' The hypothesis is that there is a correlation between language use and the level of a patient's anxiety and depression, and the measure of their self-efficacy. The second research question was '*To which extent can language used by teenage chronic fatigue syndrome patients during correspondence-based cognitive behavioural therapy be used by AI-models to predict the patient's anxiety, depression, and self-efficacy scores, as measured by self-report questionnaires, and whether they will recover from CFS through this therapy?*' It was hypothesised that the anxiety, depression, and self-efficacy and the outcome of therapy can be predicted to an extent, limited by the size of the data set.

In Section 3.3 the first research question was answered. It was concluded that there was a significant correlation between patients' anxiety, depression, and self-efficacy scores and their language use, thus confirming the hypothesis. This was concluded based on the fact that when training an AI model with randomised anxiety, depression, and self-efficacy scores, the output of the model differed in a statistically significant way from the output of a model that was trained using the anxiety, depression, and self-efficacy scores from the subjective data. The unigram model predicting the depression score gave higher accuracy scores than the models based on the subjective data.

The second research question was answered in Section 4.4. It was shown that using BERTje embeddings with a neural network to predict whether a patient recovers works fairly well, as it can be done with an accuracy of 73.1%, which is a lot higher than 61.6%, the percentage of the emails sent by patients who recovered. Anxiety, depression, and self-efficacy scores could be predicted with an accuracy higher than the models based on the subjective data, yielding accuracies of 40.8%, 38.5% and 40.5%, respectively. This is all in line with the hypothesis, as it was theorised that the size of the data set would limit the models in achieving perfect scores.

When using logistic regression, self-efficacy prediction yields more accurate results than anxiety and depression prediction. When using a neural network, this difference between all three output variables becomes smaller. Anxiety prediction even reaches a slightly higher accuracy than self-efficacy prediction when using a token size of 256. Overall, all of the neural network implementations yield accuracies that are better than the results from the models based on the subjective data.

5.1 Limitations

Some limitations of this research need to be noted. As has been mentioned before, the size of the data set is quite small. This makes it easier to overfit, as the data used could be not representative of a larger group. Additionally, as the data set was small, and the test set was only 10% of this, it could be possible that the test set is not very representative of the full data set, thus possibly influencing the results.

Secondly, the second part of this research, as described in Chapter 4, aims to predict the anxiety, depression, and self-efficacy scores taken from the subjective data. The patients might have exaggerated or downplayed their symptoms, based on what result they *want* to get from the questionnaire. This means that this research has trained models to predict values that might not have been the true values. As there was no true objective scale representative for the anxiety, depression, and self-efficacy scores used, the values from the subjective data set were used. Further research could look into other ways of determining these scores, or possibly using unsupervised learning (see Appendix C) to create classes using AI.

Another important thing to note is that the anxiety, depression, and self-efficacy scores from the subjective data set were measured at the beginning of the treatment. There is no information about these three scores at the end of FITNET. Thus, for all the emails, even if they have been sent months after the original consultation, the scores from the beginning are used. It was outside the scope of this research to check the performance of the models over time, so if emails sent at the beginning of treatment were classified correctly more often than those at the end of treatment.

5.2 Future Work

Based on this research, multiple extensions could be made. Firstly, as was said previously, other ways of defining the anxiety, depression, and self-efficacy scores could be looked at. Checking if the models still perform well with other metrics would be interesting. Additionally, using an unsupervised clustering algorithm, it would be very interesting to see if the clusters the model makes based on representations of the emails would be similar to the ones defined by the subjective data.

Secondly, it is important to revisit the way the non-agentic language features are calculated. As the distribution of these between patients who recovered from CFS and patients who did not is very similar, this goes directly against the results from Dalmaijer et al. (2021), while it is based on the same data set. Additionally, as Wignand (2021) found that catastrophizing was more common in patients with poorer treatment outcomes, it would be interesting to see how this would work as a feature. As Wignand (2021) found their results on a subset of the dataset, using the whole data set could yield interesting results.

The National Health Service (NHS) in the United Kingdom has recently implemented the FITNET program (University of Bristol, 2021). They have currently only published their pilot results, which show that they had 75 patients using FITNET (Anderson et al., 2020). It would be very interesting to analyse if similar results could be found by training models using the emails sent by the patients from this study, as those emails are in English.

Other research using this data could be done by extending the neural network models. Other types of neural network models could be used, as well as training the current model more extensively, and comparing the results. Other Dutch language models could be used as well, such as the Dutch Word2Vec model (Tulkens et al., 0023), the Dutch GPT-2 model (Vries and Nissim, 2021), or one of the other Dutch BERT models, such as RoBERT (Delobelle et al., 2020) or BERT-NL (TMR, 2022).

References

Afari, Niloofar and Dedra Buchwald (2003). “Chronic fatigue syndrome: A Review”. In: *American Journal of Psychiatry* 160.2, 221–236. DOI: 10.1176/appi.ajp.160.2.221.

Al-Mosaiwi, Mohammed and Tom Johnstone (2018). “In an absolute state: Elevated use of absolutist words is a marker specific to anxiety, depression, and suicidal ideation”. In: *Clinical Psychological Science* 6.4, 529–542. DOI: 10.1177/2167702617747074.

Al-shamasneh, Alaá Rateb Mahmoud and Unaizah Hanum Binti Obaidellah (2017). “Artificial intelligence techniques for cancer detection and classification: review study”. In: *European Scientific Journal* 13.3, pp. 342–370.

Anderson, Emma, Roxanne Parslow, William Hollingworth, Nicola Mills, Lucy Beasant, Daisy Gaunt, Chris Metcalfe, David Kessler, John Macleod, Susan Pywell, Kieren Pitts, Simon Price, Paul Stallard, Hans Knoop, Elise Van de Putte, Sanne Nijhof, Gijs Bleijenberg, and Esther Crawley (2020). “Recruiting Adolescents With Chronic Fatigue Syndrome/Myalgic Encephalomyelitis to Internet-Delivered Therapy: Internal Pilot Within a Randomized Controlled Trial”. In: *J Med Internet Res* 22.8, e17768. ISSN: 1438-8871. DOI: 10.2196/17768. URL: https://www.jmir.org/2020/8/e17768.

Azam, Saif A., Lee Myers, Brandon K.K. Fields, Natalie L. Demirjian, Dakshesh Patel, Eric Roberge, Ali Gholamrezanezhad, and Sravanthi Reddy (2020). “Coronavirus disease 2019 (COVID-19) pandemic: Review of guidelines for resuming non-urgent imaging and procedures in radiology during Phase II”. In: *Clinical Imaging* 67, pp. 30–36. ISSN: 0899-7071. DOI: https://doi.org/10.1016/j.clinimag.2020.05.032. URL: https://www.sciencedirect.com/science/article/pii/S0899707120302011.

Bandura, Albert (1977). “Self-efficacy: Toward a unifying theory of behavioral change.” In: *Psychological Review* 84.2, 191–215. DOI: 10.1037/0033-295x.84.2.191.

Berkelbach van der Sprenkel, Emma E., Sanne L. Nijhof, Geertje W. Dalmeijer, N. Charlotte Onland-Moret, Simone A. de Roos, Heidi M. Lesscher, Elise M. van de Putte, Cornelis K. van der Ent, Catrin Finkenauer, Gonneke W. Stevens, and et al. (2021). "Psychosocial functioning in adolescents growing up with chronic disease: The Dutch HBSC study". In: *European Journal of Pediatrics*. DOI: 10.1007/s00431-021-04268-9.

Bird, Hector R, David Shaffer, Prudence Fisher, Madelyn S Gould, et al. (1993). "The Columbia Impairment Scale (CIS): pilot findings on a measure of global impairment for children and adolescents." In: *International Journal of Methods in Psychiatric Research*.

Bird, Steven, Ewan Klein, and Edward Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

Bouras, N. and G. Holt (2007). *Psychiatric and Behavioural Disorders in Intellectual and Developmental Disabilities*. Cambridge University Press. ISBN: 9781139461306. URL: https://books.google.nl/books?id=E_9Rlqs4T7oC.

Bucur, Ana-Maria, Marcos Zampieri, and Liviu P. Dinu (2021). *An Exploratory Analysis of the Relation Between Offensive Language and Mental Health*. URL: https://arxiv.org/abs/2105.14888.

Coppen, P. A., W. Haeseryn, and F de Vriend (2012). *Nominalisaties*. URL: http://ans.ruhosting.nl/e-ans/14/08/body.html.

Coppersmith, Glen, Mark Dredze, Craig Harman, and Kristy Hollingshead (2015). "From ADHD to SAD: Analyzing the Language of Mental Health on Twitter through Self-Reported Diagnoses". In: *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*. Denver, Colorado: Association for Computational Linguistics, pp. 1–10. DOI: 10.3115/v1/W15-1201. URL: https://aclanthology.org/W15-1201.

Dalmaijer, Evi, Maarten van Leeuwen, and Elise van de Putte (2021). “Handelen in behandeling”. In: *Tijdschrift voor Taalbeheersing* 43.3, pp. 261–289. ISSN: 2352-1236. DOI: https://doi.org/10.5117/TVT2021.3.001.DALM. URL: https://www.aup-online.com/content/journals/10.5117/TVT2021.3.001.DALM.

de Vries, Wietse, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim (Dec. 2019). *BERTje: A Dutch BERT Model.* arXiv:1912.09582. URL: http://arxiv.org/abs/1912.09582.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2020). “RobBERT: a Dutch RoBERTa-based Language Model”. In: *CoRR* abs/2001.06286. arXiv: 2001.06286. URL: https://arxiv.org/abs/2001.06286.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (June 2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

Ehteshami Bejnordi, Babak, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, and the CAMELYON16 Consortium (Dec. 2017). “Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer”. In: *JAMA* 318.22, pp. 2199–2210. ISSN: 0098-7484. DOI: 10.1001/jama.2017.14585. eprint: https://jamanetwork.com/journals/jama/articlepdf/2665774/jama\

_ehteshami_bejnordi_2017_oi_170113.pdf. URL: https://doi.org/10.1001/jama.2017.14585.

Fukuda, Keiji (1994). "The chronic fatigue syndrome: A comprehensive approach to its definition and study". In: *Annals of Internal Medicine* 121.12, p. 953. DOI: 10.7326/0003-4819-121-12-199412150-00009.

GGZ, Nederland (2019). *Factsheet Wachttijden. Achtergrond over wachttijden in de geestelijke gezondheidszorg.* URL: https://www.denederlandseggz.nl/getmedia/af916c4f-e163-4938-818e-55b5b3cfdfdf/GGZNL_GGZ007_factsheet_wachttijden_WEB_juli2019.pdf?ext=.pdf.

Gilbert, Paul (2006). In: *Psychotherapy and counselling for Depression.* SAGE, 14–15.

Grondelaers, Stef, Dirk Geeraerts, and Dirk Speelman (2007). "A case for a cognitive corpus linguistics". In: *Methods in Cognitive Linguistics*, 149–169. DOI: 10.1075/hcp.18.12gro.

Gulshan, Varun, Lily Peng, Marc Coram, Martin C. Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, Ramasamy Kim, Rajiv Raman, Philip C. Nelson, Jessica L. Mega, and Dale R. Webster (Dec. 2016). "Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs". In: *JAMA* 316.22, pp. 2402–2410. ISSN: 0098-7484. DOI: 10.1001/jama.2016.17216. eprint: https://jamanetwork.com/journals/jama/articlepdf/2588763/joi160132.pdf. URL: https://doi.org/10.1001/jama.2016.17216.

Hong, Simin, Anthony Cohn, and David Crossland Hogg (2022). "Using Graph Representation Learning with Schema Encoders to Measure the Severity of Depressive Symptoms". In: *International Conference on Learning Representations.* URL: https://openreview.net/forum?id=OtEDS2NWhqa.

Jurafsky, Dan and James H. Martin (2014). “Part-of-Speech Tagging”. In: *Speech and language processing.* Second. Pearson Education.

Kohlberger, Timo, Yun Liu, Melissa Moran, Po-Hsuan Cameron Chen, Trissia Brown, Jason D. Hipp, Craig H. Mermel, and Martin C. Stumpe (2019). “Whole-Slide Image Focus Quality: Automatic Assessment and Impact on AI Cancer Detection”. In: *Journal of Pathology Informatics* 10.1, p. 39. ISSN: 2153-3539. DOI: https://doi.org/10.4103/jpi.jpi_11_19. URL: https://www.sciencedirect.com/science/article/pii/S2153353922004023.

Nijhof, Sanne L., Gijs Bleijenberg, Cuno S. P. M. Uiterwaal, Jan L. L. Kimpen, and Elise M. van de Putte (2012). “Effectiveness of internet-based cognitive behavioural treatment for adolescents with chronic fatigue syndrome (FITNET): A randomised controlled trial”. In: *The Lancet* 379.9824, 1412–1418. DOI: 10.1016/s0140-6736(12)60025-7.

Nijhof, Sanne L., Kimberley Maijer, Gijs Bleijenberg, Cuno S. P. M. Uiterwaal, Jan L. L. Kimpen, and Elise M. van de Putte (2011). “Adolescent chronic fatigue syndrome: Prevalence, incidence, and morbidity”. In: *PEDIATRICS* 127.5. DOI: 10.1542/peds.2010-1147.

Orben, Amy, Livia Tomova, and Sarah-Jayne Blakemore (2020). “The effects of social deprivation on adolescent development and Mental Health”. In: *The Lancet Child &amp; Adolescent Health* 4.8, 634–640. DOI: 10.1016/s2352-4642(20)30186-3.

Oren, Ela, Naama Friedmann, and Reuven Dar (2016). “Things happen: Individuals with high obsessive–compulsive tendencies omit agency in their spoken language”. In: *Consciousness and Cognition* 42, 125–134. DOI: 10.1016/j.concog.2016.03.012.

Pennebaker, James W, Matthias R Mehl, and Kate G Niederhoffer (2003). “Psychological aspects of natural language use: Our words, our selves”. In: *Annual review of psychology* 54.1, pp. 547–577.

Sharpe, M. C., Archard, L. C., Banatvala, J. E., Borysiewicz, L. K., Clare, A. W., David, A., Edwards, R. H., Hawton, K. E., Lambert, H. P., and Lane, R. J. (1991). "A report–chronic fatigue syndrome: Guidelines for research". In: *Journal of the Royal Society of Medicine* 84.2, 118–121. DOI: 10.1177/014107689108400224.

Slotman, Ellen, K Schreuder, Tamar EC Nijsten, Marlies Wakkee, L Hollestein, A Mooyaart, Sabine Siesling, and MWJ Louwman (2022). "The impact of the COVID-19 pandemic on keratinocyte carcinoma in the Netherlands: trends in diagnoses and magnitude of diagnostic delays". In: *Journal of the European Academy of Dermatology and Venereology* 36.5, pp. 680–687.

Sohl, Stephanie J. and Fred Friedberg (2008). "Memory for Fatigue in Chronic Fatigue Syndrome: Relationships to Fatigue Variability, Catastrophizing, and Negative Affect". In: *Behavioral Medicine* 34.1. PMID: 18400687, pp. 29–38. DOI: 10.3200/BMED.34.1.29-38. eprint: https://doi.org/10.3200/BMED.34.1.29-38. URL: https://doi.org/10.3200/BMED.34.1.29-38.

TMR (2022). *Language Resources by TMR.* URL: http://textdata.nl/.

Tulkens, Stephan, Chris Emmery, and Walter Daelemans (23). "Evaluating Unsupervised Dutch Word Embeddings as a Linguistic Resource". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016).* Ed. by Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis. Portorož, Slovenia: European Language Resources Association (ELRA). ISBN: 978-2-9517408-9-1.

University of Bristol (2021). *FITNET-NHS Study - Overview.* URL: https://www.bristol.ac.uk/academic-child-health/research/research/cfsme/fitnet-nhs/fitnet-nhs/.

van Eynde, Frank (2004). *Part of Speech Tagging en Lemmatisering van het Corpus Gesproken Nederlands*. URL: https://lands.let.ru.nl/cgn/doc_Dutch/topics/version_1.0/annot/pos_tagging/tg_prot.pdf.

van Noord, Gertjan (Apr. 2006). "At Last Parsing Is Now Operational". In: *Actes de la 13ème conférence sur le Traitement Automatique des Langues Naturelles. Conférences invitées*. Leuven, Belgique: ATALA, pp. 20–42. URL: https://aclanthology.org/2006.jeptalnrecital-invite.2.

Verhagen, Arie (1992). "Praxis of linguistics: Passives in Dutch". In.

Vries, Wietse de and Malvina Nissim (2021). "As Good as New. How to Successfully Recycle English GPT-2 to Make Models for Other Languages". In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics. DOI: 10.18653/v1/2021.findings-acl.74. URL: https://doi.org/10.18653%2Fv1%2F2021.findings-acl.74.

Vrije Universiteit Brussel, Universiteitsbibliotheek (n.d.). *Stopwoorden*. URL: https://biblio.vub.ac.be/noodweb/opac/stopwoorden.htm.

Wignand, Jantine (2021). *Linguistic correlates of catastrophizing in adolescents with chronic fatigue syndrome: An internship report.*

Wille, Nora, Susanne Bettge, Hans-Ulrich Wittchen, and Ulrike Ravens-Sieberer (2008). "How impaired are children and adolescents by mental health problems? results of the bella study". In: *European Child &amp; Adolescent Psychiatry* 17.S1, 42–51. DOI: 10.1007/s00787-008-1005-0.

Wu, Nan, Jason Phang, Jungkyu Park, Yiqiu Shen, Zhe Huang, Masha Zorin, Stanisław Jastrzębski, Thibault Févry, Joe Katsnelson, Eric Kim, Stacey Wolfson, Ujas Parikh, Sushma Gaddam, Leng Leng Young Lin, Kara Ho, Joshua D. Weinstein, Beatriu Reig, Yiming Gao, Hildegard Toth, Kristine Pysarenko, Alana Lewin, Jiyon Lee, Krystal

Airola, Eralda Mema, Stephanie Chung, Esther Hwang, Naziya Samreen, S. Gene Kim, Laura Heacock, Linda Moy, Kyunghyun Cho, and Krzysztof J. Geras (2019). *Deep Neural Networks Improve Radiologists' Performance in Breast Cancer Screening.* URL: https://arxiv.org/abs/1903.08297.

A Higher Order N-grams

Higher order n-grams language models work similarly to those based on unigrams (explained in Section 3.2.1.1. However, there is a difference, as a unigram is simply a word, but a bigram, or a second-order n-gram, consists of two tokens, the history and the new word. To use the same example as in Section 3.2.1.1, the sentence ‘Anna walks to school’ contains 5 bigrams, these being ‘<s> Anna’, ‘Anna walks’, ‘walks to’, ‘to school’ and ‘school </s>’, where ‘<s>’ and ‘</s>’ denote start and end tags for a sentence, respectively. Now, to calculate the probability of a certain bigram appearing, the calculation changes slightly. For a unigram, this calculation is $\frac{Count(w)}{N}$, where w is the word, and N the total number of words. For higher order n-grams, this is as follows: $\frac{Count(w_{i-1},w_i)}{Count(w_{i-1})}$, where w_i is the i-th word and w_{i-1} is the word before that. Thus, the probability of a word occurring becomes dependent on its history.

B Mean Squared Error (MSE)

Mean Squared Error (MSE) is a way of determining how well a model in a classification task is performing. It is used when classifying data where the classes are cardinally organised, which is where it is a sequence. The Error in the name MSE stands for how far from the correct label is from the predicted label. Thus, if the model predicts 2, but the correct label was 8, then the error is $8 - 2 = 6$. For MSE, this error rate then gets squared, thus resulting in $6^2 = 36$. For all the training data, it then calculates this squared error, and calculates the mean of all of these, hence the name Mean Squared Error.

C Supervised and Unsupervised Learning

There are two different types of AI classification models, based on supervised or unsupervised learning. All the models used in this paper are based on supervised learning. This means that the data that the model is trained on already has the correct output label for each data point. These labels in this research are the anxiety, depression, and self-efficacy scores, and the recovery rates of the patients. Thus, when training the model, it learns to output these values, as it knows which answer is the correct one.

Unsupervised learning is a strategy used when no such predetermined labels exist. This can be used to find clusters in large groups of data automatically, without any inference from humans.

Both supervised and unsupervised learning can be used in different applications, and have different pros and cons depending on these applications.