Universiteit Utrecht

DUTCH STYLE REPRESENTATIONS:

THEIR CREATION AND EVALUATION

RUBEN KOLE

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN ARTIFICIAL INTELLIGENCE

GRADUATE SCHOOL
OF NATURAL SCIENCES
MASTER OF
ARTIFICIAL INTELLIGENCE
UNIVERSITEIT UTRECHT

STUDENT NUMBER

8477159

DAILY SUPERVISOR

Anna Wegmann, MSc

FIRST SUPERVISOR

Dr. Dong Nguyen

SECOND SUPERVISOR

Dr. Marijn Schraagen

LOCATION

Utrecht Universiteit
Graduate School of Natural Sciences
Department of information &
computing science
Utrecht, The Netherlands

DATE

December 11, 2024

ACKNOWLEDGMENTS

First of all I would like to thank my supervisors Anna and Dong for all the hours of meetings we have had and for the countless times they gave me feedback on my writing. I would also like to thank my second supervisor Marijn for the extensive feedback on my proposal.
Next, I would like to thank my AI friends Ji, Jip, Lennard, and Raquel with whom I had the pleasure of simultaneously writing our theses. Apart from them, my other study friends, who did not always understand what I was doing, Arne, Linde, Rosa, and Wessel also deserve praise.
Furthermore, I would like to thank Arne and Linde for the ungodly liters of coffee they provided me with over the last nine months.
The last person I would like to acknowledge is my personal dictionary Ivo, who has combed this paper on grammar mistakes with the utmost dedication.

DUTCH STYLE REPRESENTATIONS:

THEIR CREATION AND EVALUATION

RUBEN KOLE

Abstract

Currently, there are no models trained specifically trained on the creation of representation of Dutch linguistic style. Neither has a task been developed to evaluate and verify the created embeddings. In this thesis, I construct a model that creates a style representation for Dutch and I create evaluation data to test if the created representation truly represents style. To create these embeddings, RobBERT-base is fine-tuned using the contrastive authorship verification task. To find the best-performing model, two datasets are constructed, and the loss function is experimented with as well as the value for the margin. The performance of the fine-tuned models falls in line with the results that are found in similar research for English style. For the evaluation, the STEL dataframe is adapted to a Dutch version. Some categories are copied from the English variant and translated to properly reflect Dutch style. Other categories are novel in this version. There are two versions of the STEL task, one of which controls for content to ensure that the embedding makes a decision based on style. The performance of the embeddings on the STEL task shows similarities to the results that are found in research into the English equivalent and shows that for most tasks the fine-tuned model learns to perform better on the tasks that control for style than the baseline model does. Therefore, this thesis concludes that it is possible to utilize methods devised for creating and evaluating English style representations and transform these into a Dutch version that show similar results as the original do.

CONTENTS

1 INTRODUCTION

It is hard to pinpoint what exactly constitutes linguistic style, which has been a topic of research since at least 1969 (Crystal & Davy, 1969). Since at least then, researchers have failed to reach a consensus on the definition, with various scientific fields adopting different approaches to define it (Herrmann, van Dalen-Oskam, & Schöch, 2015). Though researchers cannot agree upon a universal definition, the importance of style cannot be understated. It can be an expression of identity for an individual or for a group (Khalid & Srinivasan, 2020; Zhu & Jurgens, 2021). An example would be that the posh high-class aristocrats have a different linguistic style than blue-collar workers do. The lack of a universally recognized definition, combined with the prevailing presence of style, raises the question of how machines could handle style in texts. For the English language, much research has been conducted into the use of machine learning techniques that look at style. A collective overview can be found in Savoy (2020, p.109-148). Examples are K-nearest neighbors, support vector machines, and logistic regression.

Additionally, extensive research has been conducted on the representation and encoding of style (Toshevska & Gievska, 2022; Wegmann, Schraagen, & Nguyen, 2022). Style generally refers to the manner in which something is communicated, rather than the content of the communication itself. As a result, research methods for identifying textual style range from simple approaches that analyze stylistic features to advanced machine learning models that learn style representations. Texts are presented to a model in an encoded, numerical format called a representation. This is done to help the model process and interpret the data more effectively. This thesis will explore the way Dutch linguistic style can be represented using text representations for language models.

This thesis aims to build a representation of Dutch linguistic style and create a suitable method to evaluate this representation. For English style, there are existing evaluation methods, e.g. the STEL framework from Wegmann et al. (2022). These evaluation tasks are designed to verify if the representation truly depicts style or if the model created a representation that did not learn to do so. The problem is that it is hard to determine if two sentences are classified as having a different style, because their style is different or if there is another factor at play; it could be the case that the model is looking at other features of the text instead of style, such as the topic. Since there has been significantly more research conducted into English style compared to Dutch style, this thesis will take this research as a base and use this to build on their methods to create a Dutch equivalent. This is both the case for the creation of the model as is it for the creation of the evaluation.

1.1 *The omnipresence of style and its uses*

Linguistic style, hereafter referred to as *style*, is present in every sentence, whether it is put there consciously or unconsciously (Crystal & Davy, 1969). The style of a text is shaped by its author. Style can occur on multiple levels. For example, the formality of a text can be classified as a stylistic feature of a text. At the same time, the number of misspelled words can also be seen as a stylistic feature of a text (Holmes, 1998). Besides this, style is also the choice of words and the order in which those words are written down. Although many examples of what style is can be listed, there is no agreed-upon operationalization for defining style due to its inherent vagueness and the lack of a universally accepted definition. All in all, style is omnipresent in language, whether we actively acknowledge it or not.

Due to its pervasive nature and the extensive research dedicated to defining it, style has become a distinguishable feature in many texts. Characteristics of a writer's style can usually be distinguished either manually (Fedulenkova, 2018) or computationally (Neal et al., 2017). There are of course exceptions. e.g. when the author actively tries to change the style of their writing (Neal et al., 2017) or when the available messages are extremely short it might also not be possible to distinguish style (Grant, 2013). This characterization can contribute to uncovering whether two texts have the same author, as a machine learning model expects texts of an author to exhibit similar style elements in all their texts. The authorship verification task, a task to determine if two pieces of text are from the same author, is often used in forensic analysis. For example, fraudulent messages sent to multiple victims could be traced back to one single sender. It can be used for prosecuting terrorists, murderers, and scammers in a court of law (Abbasi & Chen, 2008; De Vel, 2000). Additionally, in academic journals, authorship verification is used to ensure that no academic fraud is committed (Iqbal, Khan, Fung, & Debbabi, 2010). And, on social media, accounts can be linked back to being from the same user when policy violations are encountered (Boenninghoff, Nickel, Zeiler, & Kolossa, 2019). This task frequently uses the style of texts to identify whether there is a common author behind texts posted from different accounts. (Sundararajan & Woodard, 2018).

Besides authorship verification, there are other tasks for which style is important. An example is style transfer (Hu, Lee, Aggarwal, & Zhang, 2022). This is a task where the style of a certain piece of text needs to be transferred to another text or a task where the content of a document needs to remain the same, but the style needs to be changed. Another task that can utilize this representation is linguistic style accommodation: matching the style used by a conversation partner can benefit the grounding of the conversation and create a better understanding with the other speaker (Danescu-Niculescu-Mizil, Gamon, & Dumais, 2011). These tasks benefit greatly from utilizing these representations as a component of the data. This is often in the form of a representation that represents words, sentences, or even whole texts as vectors. The representation created for one task might also be used for the other since both need style representation to function well (Fu, Tan, Peng, Zhao, & Yan, 2018).

1.2 *Style representations*

The goal of this thesis is to train a model that creates embeddings for linguistic style in Dutch and evaluates those style embeddings. The created representations are in the form of text embeddings. Text embeddings are a specific type of representation that functions by representing a text as a series of numbers. These numbers correspond to a position in an N-dimensional space. The goal is to place sentences that have matching styles close together in this space while sentences that differ greatly in style are not near each other. This means that sentences such as «*Thank you for your time; I look forward to our next meeting.* » and «*It was a pleasure* *speaking with you; I wish you all the best.* » would be close together in this space. However, «*It was fun seeing you. Catch you later!* » would be far removed from the other two sentences. This should be regardless of what the text is about.

A simplified example would be a binary vector that assumes the value 1 for the presence of a feature and 0 if this feature is not present. Features can be spelling mistakes, the use of exclamation marks, or any other style-related aspect. Transforming texts into representations allows the texts to be compared easily with other texts based on what the representation was made to represent. The key difference between the given example and the representations constructed in this thesis is that the features of the representation are not explicitly known because of the neural nature of the model. This also means that we cannot be certain of what the model created in this research exactly represents. To assess this, evaluations are required. These evaluations look at specific style aspects and verify if the model recognizes the difference between sentences that have different writing styles. If a model can do this, it does not guarantee that the model is also able to distinguish texts from different authors. However, it is the case that every author has their writing fingerprint, also called idiolect. This idiolect might be hard to pick up for humans.

Research into more broadly usable representations that can be used for multiple style tasks has been conducted in English (Hay, Doan, Popineau, & Ait Elhara, 2020). For the Dutch language, this research is close to non-existent. Some literature suggests that style models can be created in such a way that they are usable for authorship verification tasks in all Indo-European languages (Adamovic, Miskovic, Milosavljevic, Sarac, & Veinovic, 2019).

In English, the most successful types of models for constructing representations are BERT-type (Devlin, Chang, Lee, & Toutanova, 2019) models. These models are designed to create representations for texts. Their main focus is on semantics and not on style. There also exist BERT-type models in Dutch, such as RobBERT (Delobelle, Winters, & Berendt, 2020). The representation that they create can be evaluated in different ways. For this thesis, an existing framework for English style representation evaluation (Wegmann et al., 2022) will be adapted for Dutch. This framework is called STEL and functions by creating sentence pairs in which the content of the sentences is the same but the style of the sentences differ on a specific stylistic dimension.

1.3 *Research questions*

So, there is little research about style representations in Dutch, while this is important to be able to represent in models. At the same time, there is research that attempts to tackle this very problem in the English language. Therefore, this thesis sets out to create a representation of Dutch linguistic style together with a Dutch evaluation dataset that can be used to evaluate the representations. This leads to the following research question:

> *How can we translate English approaches for the training and evaluation of style representations?*

To answer this question, the representations will be created by finetuning a BERT-like language model per prior research (Wegmann et al., 2022). The evaluation dataset will be constructed based on prior research in English (Wegmann & Nguyen, 2021) combined with new dimensions unique to the Dutch dataset. Both papers investigate their respective problems for the English language. This thesis will adapt their methods to Dutch data. To accomplish this, the following subquestions have been devised:

RQ 1. How can the methods of Wegmann et al. (2022) be adapted to fine-tune a model to learn Dutch style representations?

RQ 2. How can the evaluation framework by Wegmann and Nguyen (2021) be adapted to evaluate Dutch style representations?

Research question 1 focuses on the first step of this research, creating a model that can construct a representation with the help of the Dutch version of Wegmann et al. (2022)'s authorship verification task. Answering this question is crucial, as the model's performance on the task serves as a strong indicator of the quality of the representations it generates. The task of creating style representations has already been used in research focused on the English language (Wegmann et al., 2022), but no research has attempted to create these representations for Dutch. BERT-like language models have been created to construct representations for Dutch (Daelemans, 2013; de Vries et al., 2019; Delobelle et al., 2020), but these are not focused on style. The challenge will be to combine knowledge from English research about style representations with research about Dutch representations. The model created in this thesis is trained to differentiate authors based on their style. This training task ensures that the model can see the difference between authors' texts. The evaluation then sees if the performance of the model is thanks to its recognition of style or if there is another factor that plays into the decision-making process.

After the creation of the representation, it is necessary to create the evaluations for the representations. Research question 2 is devised to answer the question of how existing English evaluation methods should be transferred to Dutch. The dataset that will be adapted, contains tasks in which four sentences are presented. There are two sentences written in a particular style and two are written in another style. The content of the sentences is as much the same as possible.

1.4 *Overview of this thesis*

To answer these questions, this thesis consists of 5 parts. The first, chapter 2, is the literature background in which the main concepts will be discussed thoroughly. These are style, authorship verification, style representation, style models, and lastly, the research that has been conducted into these domains in Dutch. Thereafter, chapter 3, the methodology chapter contains information about the methodology for this thesis. In the results chapter 4, the results are discussed. In chapter 5 the results are discussed and explained. Lastly, there is chapter 6 where this thesis' findings are shortly summarized

2 LITERATURE BACKGROUND

In this section, the scientific background of key aspects will be discussed. The goal of this thesis is to evaluate created representations of Dutch style. Therefore, understanding what constitutes style plays and important part of this thesis. Section 2.1 will discuss the definitions of both factors of style that could be detected by machines, humans, or both. This section will also examine the factors of stylometry and stylistics, along with their relevance. After gaining familiarity with the terminology of style, section 2.2 will discuss how the authorship verification task can be used to build a representation. Then, section 2.3 will highlight research into representations. Finally, the differences between Dutch and English style research will be discussed in section 2.4.

2.1 *Style*

Derived from the 14th-century word *stile*[1], the writing instrument of an author, style is nowadays described as «a way of doing something, especially one that is typical of a person, group of people, place, or period»[2]. Style can be a tool for an individual to express themselves and stand out from others. Everyone who writes a text has their own way of doing so. The choices made in this process, whether conscious or unconscious, are the style of a text. In this thesis, only some facets of style will be used. Examples are informal or formal style and the use of punctuation marks. These features are examples of stylistic features; however, since a model is not explicitly informed about them, it is uncertain whether a model will learn to identify them. This subsection about style will first examine the different interpretations of the term style in section 2.1.1. Secondly, stylistic and stylometric methods will be compared in section 2.1.2 since these are needed to manually and machinally find style in a text. Thirdly, section 2.1.3 will discuss the concept of individual style, also called idiosyncrasy.

2.1.1 *Style in language*

Speakers may make choices regarding vocabulary and grammar based on non-linguistic factors, such as the purpose of their communication or the relation between themselves and the listener (Biber & Conrad, 2019). These choices can be categorized under the umbrella term **linguistic style**. This section will delve into what this linguistic style exactly constitutes.

Definitions of linguistic style. As previously mentioned, the term style has been conceptualized in many ways. In the 1960s, descriptive linguists used the term to refer to general situational varieties (Joos, 1967). This means that when listening to an excerpt of a conversation, most people will be able to apprehend the relation between the conversation's participants without being given any context before listening. On top of this, by assessing the language used in the presented excerpt, almost all will be able to make the right assumption of the relation between the two speakers (Crystal & Davy, 1969). An example of this

[1] https://www.etymonline.com/word/style

[2] https://dictionary.cambridge.org/dictionary/english/style

would be overhearing a parent talking to their baby. The language used would be distinct enough to derive the relation between the speakers. This example merely credits style with the change in vocabulary and grammar based on a conversation's participants. However, style can be attributed to more functions in linguistics.

For example, quantitative sociolinguists such as Labov used the word style to accredit the language used for different purposes in a sociolinguistic interview to style (Labov, 1975, 1985). This refers to the change in someone's use of language based on the attention needed to complete a speech-related task. If a task was more difficult it forces the participants to use a more authentic style. Therefore there was a large difference in style used during easy and difficult tasks.

Definition of style for this thesis. For the remainder of this thesis, I will assume the following definition of style: *Style is a property of texts constituted by an ensemble of formal features that can be observed quantitatively or qualitatively.* (Herrmann et al., 2015). The formal features refer to linguistic features at the character, lexical, syntactical, and semantic level (Stamatatos, 2009). While the chosen definition of style is quite broad, it captures what is required for this thesis, it describes style in such a way that a machine can infer style by observing features quantitatively. At the same time, it allows me to be able to test if the model observed these features correctly by testing it using qualitative features such as the tone of a text.

2.1.2 *Stylometric and stylistic methods of analyzing style*

Discovering style in a piece of text can be achieved by manually looking at a text and determining pieces of text that are distinctive linguistic features that reflect an individual's unique use of language called style markers (McMenamin, 2002). based on this, a machine can look at specific statistics of a text and determine the style of a text based on these. These are referred to as stylistic and stylometric methods respectively. While both can lead to correct authorship verification, some differences should be highlighted.

Stylistics. Before the use of algorithms, there was the need to rely solely on the findings of linguistic experts. In the field of authorship verification, this was presented as stylistics. Stylistics is the part of linguistics that is concerned with style in spoken and written language, especially the word choiceds and syntactic structure (Leech & Short, 2007). It dates back to the notion of rhetoric, the study of how to make speech more compelling (Giovanelli & Mason, 2017). The stylistic method can be defined as the «application of the science of linguistic stylistics to forensic contexts and purposes» (McMenamin, 2002, p. xii). Even though this quote only speaks of forensics, it still applies to other fields where linguistics is present.

As stated before, stylistic methods will be needed in this thesis to assess the model's representation of style. These will be used to verify the results the model has reached. Therefore, some ground rules about stylistics need to be laid down. Stylistics must be rigorous, retrievable, and replicable (Fedulenkova, 2018). Being rigorous means that the researcher does not nitpick certain features in a text and instead looks at the text holistically. It is hard for an expert to look at the whole text and to consider all aspects of the text to make a decision. A conclusion that is retrievable is a conclusion where it is clear to the interpreter why a certain conclusion was reached. This is achieved by finding markers that have a solid academic basis on which their usage is justified. Lastly, there is replicability which means that the conclusions reached can be replicated by others when they are using the same methods.

Stylometry. At the foundation of modern stylometry are Zipf and Yule (Yule, 1925) who made the quantification of features in text mainstream (i.e. Zipf's law (Kingsley Zipf, 1932)). At the basis of computer-assisted stylometry are Mosteller and Wallace with their research into the Federalist papers (Mosteller & Wallace, 1984) in which hey applied stylometry to the 12 essays of which the authors were disputed. Hereafter, the invention of digital technologies such as machine learning, information retrieval techniques, and neural networks led to an improvement in natural language processing techniques (Matthews & Merriam, 1993; Merriam & Matthews, 1994; Tweedie, Singh, & Holmes, 1996).

Examples of stylometric methods are counting N-grams or word frequencies (Holmes, 1998). An N-gram is a sequence of N items (such as words or characters) from a text. Word frequencies and n-grams are statistics that could hypothetically identify an author by looking at the details of a text. Another approach that can be utilized is looking at the text holistically and selecting specific examples to be used as evidence for identifying a certain author. However, it could be argued that the whole text is still taken into consideration when looking at word frequencies, since these occur spread over the whole text. That would mean that those techniques are applicable as well.

A problem with stylometry is that authors can change their writing style, making the identification task harder (Neal et al., 2017). Additionally, the authors of the paper ask the question of whether stylometry techniques are generalizable across genres and topics and confirm Daelemans (2013)'s report in which he states that this is only possible when there is a large balanced dataset.

All in all, stylometry relies on statistical patterns in a text instead of solely relying on the researcher to look for markers in a text. This makes texts to be investigated in a more objective manner, but it might also lead to a less explainable conclusion since the exact rationale that a human uses might not be deducted from the parameters that a machine provides. To combat this, stylistic techniques will be used to verify the results of the model.

2.1.3 *Idiosyncrasy*

Style can thus be seen as something that is representative of the way a certain group uses language. Another way to look at it is how the language use of an individual differs from that of others. Idiolect refers to the style used by a singular person. Even though this concept is prominent in linguistics, there is no consensus about what exactly constitutes idiolect (Wright, 2018). It can be classified as all the possible combinations of conveying a message (Bloch, 1948; Turell, 2011), or it can be described as «a language that can be characterised exhaustively in terms of intrinsic properties of some single person at a time» (Barber, 2008).

By contrasting the writings of authors whose background demographics were strictly controlled, the existence of an idiolect was tried to be shown (Kredens, 2003). If these people were to style their stories differently despite their similar backgrounds, then the explanation must be that there is some individuality in the writing that is not accounted for by the upbringing of the writers.

A limitation is that a large quantity of text might be needed to detect an author's idiolect (Olsson, 2007; Turell, 2011). In the field of forensic linguistics, there is not always a large availability of long texts to gather evidence or perform authorship verification. These studies showed that it is possible to perform this task with short messages (Amos, 2008; Grant, 2013).

2.2 *Authorship verification using style*

Style is an important part of one's written text. It allows for a model to recognize aspects of style and make a representation based on these aspects. In essence, authorship verification is the task of determining whether or not two texts are written by the same author. In the verification task, the authors' idiolect is of great importance. As mentioned before in section 2.1.3, the idiolect is the unique writing style of the author and it allows the model to make a distinction between authors. Finding these styles helps the model perform authorship verification (Howard, 2009) and it helps with the building of the representation since it allows the model to distinguish between styles.

In the current subsection, the notions of authorship verification and contrastive authorship verification will be laid out. Thereafter, the idiolect indicative of individuals will be sought with the use of style markers. They will be used as the building blocks on which models base their decisions. These markers will also be used to construct the embedding of the model. Finally, the role of text size and computer-mediated communication will be discussed. These aspects are of importance because the data that will be used in the experiment originates from sources that contain texts of varying lengths from an online medium.

2.2.1 *Authorship verification & authorship attribution*

A stylistic representation cannot be extracted from a model by simply giving a model a text. It needs a task on which it can be trained and through which it learns what constitutes style. In this thesis, the task will be a form of authorship verification. This task makes the model decide if two texts are written by the same author or by different authors (Gao, Yao, & Chen, 2021; Wegmann et al., 2022).

Authorship verification. Authorship verification (AV) aims to determine if two texts have the same author. This goal has been tried to be attained for decades. One of the first recorded instances was in 1581 when Augustus de Morgan used the length of words to compare texts (Olsson, 2007). Then, in 1887 Mendenhall published about the frequency distributions of authors in different languages (Mendenhall, 1887). In 1901 he published a study that theorized that the frequency distribution of word lengths in Shakespeare's plays was significantly different than the frequency in the writings of Bacon. He claimed it would be impossible not to be able to distinguish the two based on this frequency (Mendenhall, 1901). Even though a methodological mistake led to the discredit of Mendenhall's findings (Williams, 1975), his attempts are still the basis of authorship verification research.

This basis of modern authorship verification is based on stylometric methods, as described in section 2.1.2. While the roots of stylometry are simple, meaning that they were based on counting words or character n-grams (Mosteller & Wallace, 1984), the approaches that were used became more advanced with time. With the introduction of machine learning, k-nearest neighbors (KNN) was implemented to find authors that were closest to new texts (Halvani, Steinebach, & Zimmermann, 2013). Here, the style features were handpicked and sorted into categories. Then the frequencies of the features were counted in the documents and based on these frequencies the documents were embedded. Support vector machines (SVM) were also used to solve the problems of authorship attribution (Demir, 2016; Diederich, Kindermann, Leopold, & Paass, 2003) and authorship verification (Adamovic et al., 2019). Neural networks then presented new ways to perform the tasks (Litvak, 2019). However, this did cause the models to be less transparent in their decision-making process.

Authorship attribution. The roots of the authorship verification task lie in the authorship attribution (AA) task. The task consists of attributing texts to authors from a selective pool. For example, contemporary poets. The similarities between AA and AV are that both tasks require the comparison of texts. The attribution task asks that a text be attributed to one author out of a range of possible authors. This means that this task is less general and significantly harder since it is no longer a binary decision, but a multiclass decision instead. When using AA, there is less content control possible since it would be hard to specifically train the model to distinguish the authors purely based on their style without focusing on content. This can be done in an AV task using contrastive authorship verification, this task will be discussed in section 2.3.2.

2.2.2 *Authorship features*

To detect the style in a text, there need to be elements that are indicative of this style. These building blocks of style are called style markers (McMenamin, 2002; Smith, Spencer, & Grant, 2009). As their name suggests they are features that mark style. These style markers can be used to verify whether a model is truly making a decision based on style. If it is, then the embeddings of a text should change accordingly with a change in style markers.

Markers of style in a text can be observed across linguistic levels (Stamatatos, 2009). These levels range from the smallest units of language to broader contextual elements. For instance, at the smallest level, features like letter frequencies might be considered. Moving to a higher level, variations in word formation could serve as a marker. Higher levels might involve sentence structures or patterns, while semantics could focus on the nuances of meaning within phrases. This layered approach demonstrates that stylistic information permeates all aspects of language (Smith et al., 2009)

Other researchers have opted to not use the linguistic layers but instead use feature categories. Examples of these categories are punctuation marks or the use of function words (Halvani et al., 2013). These categories allow for a more precise distinction between single features. Punctuation marks would be prone to ambiguity when using linguistic layers. This new method allows them to be their own category.

Another possibility is to have a list of features that are theorized to have a stylistic effect on a text. Several lists have been made and all have other features. The most common style marker is often-occurring word frequency (Burrows, 2002; Hoover, 2001). This feature only considers common words like function words and thus excludes jargon and other less often used words since an author is likely to use the same words more often. As will be discussed in section 2.3.2, this method is chosen because these function words work independently of the topic of the text. Another common inclusion on constructed lists of features is both letter and word n-grams. These can range from bi-grams to hexagrams (Eder, 2011). An author probably structures their sentences in such a way, that they often use combinations of words together. After all, the meaning of a word is determined by the company it keeps (Firth, 1957), implying that a recurring group of words can tell more than only a single word.

2.2.3 *Factors influencing the performance of authorship verification*

A model performs better if it receives better data. There are a few factors that should be considered when choosing the data that is used for authorship verification. These are the lengths of the texts, the unique number of authors, and the number of texts per author (Yüzer, 2022).

Text size. At the foundation of stylometry, there were suggestions that the minimum length of texts should be 500 words to get a reliable result out of an AA task (Forsyth & Holmes, 1996). This was performed as an AA task because AA tasks used to be the main task whereas AV used to play a minor role. The 500-word limit was refuted a few years after Forsyth Holmes' research when it was discovered that tasks with shorter texts were also performed relatively well compared to longer texts (Argamon, Koppel, Pennebaker, & Schler, 2009). Real developments in the realm of short texts started occurring when studies started to focus on small texts thanks to the rise of the internet (Abbasi & Chen, 2005). A study with a 97% success rate and an average text length of 77 words showed that it was possible to use shorter texts (Abbasi & Chen, 2008).

Twitter used to have a maximum message length of 140 characters, providing a useful source for short messages. When using these messages, authorship verification achieved promising results on authorship verification with tweets, however, they needed a minimum of 120 messages per author (Layton, Watters, & Dazeley, 2010), which presents the next challenge.

Texts per author. Ensuring there is sufficient data per author to be able to differentiate between them is a challenge. If an author only has a few texts, it will be hard to recognize the style markers that define that author. Research, where there are many authors with a limited number of texts to their name, points out that this set-up yields results that are inferior to other studies that have a different distribution of texts per author (Luyckx & Daelemans, 2008). This is mainly a problem in authorship attribution and less in verification. Because for verification, pairwise only examples are needed, meaning that less data is needed to differentiate between authors.

Unique authors. Apart from the lengths of texts, the number of authors is also important for AA tasks. This is slightly less relevant for CAV tasks since these tasks only answer a binary question instead of a multi-class problem. However, the number of authors is still an important factor to control since a larger author pool allows the model to be exposed to more writing styles allowing it to be more generalizable. In general, authorship tasks require the set of authors as well as the content of their writings to be carefully considered (Juola, 2015; Stamatatos, 2007).

All in all, during this thesis it is important to keep in mind that there are several factors at play that can influence the styles used by the authors. Some of these are caused by the style used in the texts. These circumstances cause the model to model a specific style. This could mean that the representation of this thesis will not be all-encompassing, but will instead only apply to texts produced in a certain environment. Other factors are caused by the corpus that models are trained on.

2.2.4 *Authorship via computer-mediated communication.*

The medium through which communication happens partly determines the interaction's style. Mails are often more formal than Twitter conversations or discussion forums. Since this thesis will use data gathered from Reddit, it needs to establish what the differences are between texts on forums and other texts. The structure of Computer-mediated communication (CMC) is determined by circumstantial features (Thurlow, Tomic, & Lengel, 2004): firstly, there is the type of channel and the mode of communication it offers. Almost all channels allow you to use text, but not all have a feature that includes videos or pictures as message formats. Another circumstantial feature is the structure of the conversation. Does the platform allow for direct replies, and are these replies in a thread format or simply unstructured? Lastly, there are the users of the platform. Who they are and to which demographics they belong is of importance when doing research. The demographics between platforms are significantly different.

CMC introduces new aspects of communication compared to physical written texts. For example, emotional 'noises' such as «*hahaha*» or «*owowow*» are not often used in physically published texts. In CMC, these are used more often (Crystal, 2005). Additionally, the introduction of emoticons expanded the range of ways in which users could express themselves online (Lee & Barton, 2013). In their context, emoticons were seen as a string of characters that resembled a face with a certain emotional connotation linked to the combination of characters. Research has been conducted in which emoticons were implemented as features (Altamimi, Clarke, Furnell, & Li, 2019). In their paper, the emoticons did not make the top ten most important features for authorship authentication. This does not mean that they are unimportant, but their significance cannot be determined from the statistics given by the researchers in their paper. Other research regarding authorship verification has also included emoticons, but they also did not report any definitive conclusion on their inclusion (Brocardo, Traore, & Woungang, 2015; Li, Chen, Monaco, Singh, & Tappert, 2017).

The difference between emojis and emoticons is that the latter are created by regular keyboard characters while the former are small pictorial images. Emojis have changed CMCs even further than emoticons already did (Evans, 2017). Emoji use is often ambiguous and their use often leads to miscommunication (Ai et al., 2017; Miller, Kluver, Thebault-Spieker, Terveen, & Hecht, 2017). This means that emojis form a perfect addition to the idiolect of an author. The context in which emojis are placed, tells us about the author's interpretation of said emoji, allowing the model to potentially differentiate the contexts in which emojis are used. The inclusion of emojis as a style marker proved to be fruitful (Marko, 2021). Additionally, it is stated that there might be many influences in the decision-making process regarding emojis such as preference, con- or divergence from the other's emoji usage, or even the used device (Marko, 2022).

2.3 *Style representation and evaluation*

Natural language is sparse and discrete. Still, we want models to work with this data while having a computational cost that is as low as possible. The data needs to be transformed into a more machine-readable format to do this. Most often, this is in the form of a high-dimensional vector. This allows for easier comparisons between instances of data, in this case, texts. After the creation of the representation, this representation needs to be evaluated. This section will discuss the creation and evaluation of style representations.

This section starts with section 2.3.1 in which an explanation of textual representations in general is given. It explains what they are, and how they represent words, sentences, or whole texts. Thereafter, section 2.3.2 will explain contrastive authorship verification tasks together with the importance of controlling for content when creating a representation. Then, in section 2.3.3 the creation of style representation is discussed with a focus on the models used to create the representation. Lastly, the evaluation of the representations is discussed in section 2.3.4.

2.3.1 *Textual representation*

Since transformer models cannot process words they need their input to be transformed into a numerical format before sentences can be used. When designing this transformation, it is important to keep in mind that features of all linguistic levels must be retained in the process (Gero, Kedzie, Reeve, & Chilton, 2019). A representation can only fully capture all aspects of style when information about all levels is still represented.

To illustrate what makes a representation good and what makes a representation lacking some features, a variety of options will be discussed next. These will not be used in this thesis, but they do illustrate what representations are in a basic sense, and they show how they can be used to transform sentences into machine-readable formats.

One-hot encoding. One of the easiest ways to transform a sentence into a more machine-friendly format is by applying one-hot encoding. With this method, every word is represented by a vector the length of all unique words. Every unique word gets assigned an index on this vector. Then for the word that is represented, all indexes are set to zero except for the one that matches the word that needs to be represented. Even though this method does allow machines to work with the data, the dataset that is created is sparse and computationally expensive.

TF and TF-IDF. Term frequency (TF) is based on the frequency of word occurrences. Instead of creating a largely empty matrix with only one non-zero value, this method creates a largely empty matrix with the counts of frequency of the occurring words. Every word that occurs in the collection of documents gets a unique index. Then, every sentence or document can be represented with a vector that describes the frequency of the occurring words.

The Term Frequency-Inverse Document Frequency (TF-IDF) is a method to see how often a word occurs in a text compared to the relative occurrence of that word in other documents (Jurafsky & Martin, 2024; Rajaraman & Ullman, 2011). It is most often used in information retrieval or text mining. However, it can also be used to compare texts with each other. It gives word scores. These scores can be indicative of word use by certain authors and could therefore be used to represent authors or texts.

Word embeddings. The problem with the previously mentioned methods is that they are most often sparse and lack the capability to encode relationships between words. Word embeddings are dense representations created by neural networks. These neural networks are trained on the context in which words appear. Examples of word embeddings are GloVe (Pennington, Socher, & Manning, 2014) which uses global statistics of the document to derive an embedding, and Word2Vec (Mikolov, Chen, Corrado, & Dean, 2013). This model is trained to predict a word based on the surroundings of that word or to do the opposite and predict the surrounding words for a single word. The advantage of word embeddings is that this form of representation allows for a comparison that is not necessarily based on frequency but instead focuses on the relationship between words.

The Count vectorizer and TF-IDF can represent entire documents in a single vector. This means that documents can be compared with each other with relative ease. However, word embeddings, as the name implies, only embed single words meaning that they cannot represent whole texts. Contrarily, word embeddings, though dense in structure, can only be used to compare documents meaningfully when the average word embeddings of the words in a sentence are taken. This does mean that word order is lost. What is needed is a representation that can densely represent whole documents to allow for computationally cheap comparisons between documents.

Document representations. There are multiple ways to represent a whole document. The most obvious way is to assign a score to each document based on a certain characteristic. This can be the average word or sentence length in a document or the average number of nouns. This method would lose a lot of information in the transformation. Another method would be to represent the document as a vector, as discussed in the previous section. A vector could represent the frequency of punctuation marks (Sari, Stevenson, & Vlachos, 2018) or the similarities between the frequency of all character 3-grams (Neal et al., 2017). This would result in a more extensive comparison, but there are two downsides. First of all, it requires the manual selection of features, which means that features could be chosen that have no impact or there could be features that do have an impact that are overlooked. Secondly, the comparison might be more extensive, but previous research suggests that other methods achieve better results (Wegmann & Nguyen, 2021). One such method is based on a Bidirectional Encoder Representations from Transformers (BERT) approach. This type of model will be explained after the training task it is needed for is introduced.

2.3.2 *Contrastive authorship verification & controlling for content*

Just as Word2Vec and GloVe are general representations created by training a model, a representation for documents can also be created using a similar technique. Word2Vec utilized the word prediction task, where words were masked and the model had to predict what the word would be. For style representations, a variant of the authorship verification task will be used to train a BERT-type model. In the following section, the focus will shift from general representations to the representation of style.

Whereas conventional authorship verification tasks (section 2.2.1) attempt to determine if two texts are written by the same person, contrastive authorship verification (CAV) (Gao et al., 2021) can be seen as a combination of authorship verification and authorship attribution. In a CAV task, an anchor text is presented with two other sentences. One of these two sentences has the same author as the anchor text, while the other is written by a different author. The goal is to match the texts that have the same author. This change means that the model does not only learn what to look at, but also which features from texts could indicate that two texts are not written by the same author. An example can be found in Figure 1.

Regarding the creation of representations of style, the focus on what should not be learned is relevant, since the representation ought to focus on style and not on content. That is why researchers control for content, e.g. (Wegmann et al., 2022). This is required since style is closely related to content. To illustrate the difference, a comparison between a poem and a play can be made. While both can address the same topics, they do not look alike in their presentation. In this example, the difference between style and content is clear, but there are many cases where this is different.

Another problem solved with the use of content control is that authors are likely to write about certain topics more often than other topics. When entering these authors' data in a model without considering this, there is a chance that a model will not select based on style, but merely on the content of the text. When an author uses the same jargon in separate texts, the model might pick up on this especially if other authors do not write about this specific topic. To combat

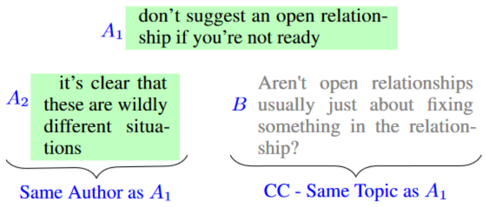

A_1
don't suggest an open relationship if you're not ready
A_2
it's clear that these are wildly different situations
B
Aren't open relationships usually just about fixing something in the relationship?
Same Author as A_1
CC - Same Topic as A_1

Figure 1: Example of a contrastive authorship verification task (Wegmann et al., 2022). The content is controlled by matching the topics of A_1 and B and by matching the author of A_1 and A_2.

this problem, a model can be trained on utterances from the same author that have a different topic. This is done by taking three utterances in total: two from one author which differ in their topic and one utterance from another author. This utterance has the same topic as one of the original author's utterances. This ensures that the model is forced to base its decision on style and not on content (Wegmann et al., 2022).

2.3.3 *Creation of representation*

Now that the task is clear, it needs to be executed by models. As mentioned, prior research has suggested that BERT-type models are well-equipped to perform the CAV task and to create representations (Wegmann & Nguyen, 2021). The goal of the model is to detect style markers and decide if two texts are written by the same author based on those markers. BERT utilizes features that differ from those used by most traditional machine learning models. While BERT relies on embeddings as inputs, 'classical' models typically work with features represented as structured data, such as weather statistics. Therefore, we cannot be certain that it will use the style markers we intend it to find. Here the differences between stylometrics and stylistics (section 2.1.2) are highlighted once again. The model, using a stylometric approach, can be checked by applying measures taken from stylistic approaches. So, as long as there are valid checks for the created representations, it is acceptable not to know exactly what the model's decisions are based on. This is not a problem as long as the evaluations can answer whether the representation is based on style.

The predecessors of BERT are similar in the way that they are language representation models specifically designed to create representations of language. These models are trained using a feature-based approach or using a fine-tuning approach. The former trains models on task-specific objectives while the latter finetunes all parameters. All of these approaches function by processing the data unilaterally. This changed with the introduction of BERT (Devlin et al., 2019).

BERT is trained in a bidirectional manner. The training task is a masked language model task. Here, one word is left out of a sentence and it is the model's goal to predict this sentence. The model takes data in front but also data occurring after the masked token as input. Using this task, BERT creates a pre-trained representation that is used for various natural language processing tasks.

Inspired by BERT, other models are built on its main concept but with some improvements. One of these spin-offs is RoBERTa. The Robustly optimized BERT approach (RoBERTa) (Liu et al., 2019) suggests that the original BERT model was undertrained and that by training the model longer with bigger batches, the performance of the model can be optimized. Additionally, new training data was added. This, with some other changes, caused RoBERTa to achieve state-of-the-art results.

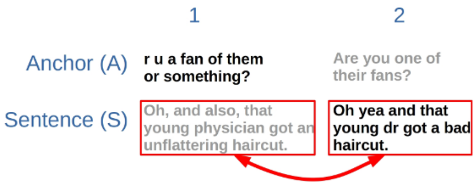
Another variant of BERT is SBERT. Sentence-BERT (SBERT) (Reimers & Gurevych, 2019) addresses the problem of the computational overhead occurring when large sequences of embeddings from either BERT or RoBERTa need to be compared. This is achieved by utilizing Siamese twin and triplet models. SBERT has also been used in previous research to create style embeddings (Wegmann et al., 2022).

2.3.4 *Evaluation of representation*

The CAV task is devised to force the model to only focus on style, but there cannot be a guarantee that the model will focus on this or that it is still somewhat focused on the content of the text. The goal is to find a way in which it can be verified that the representation is focusing on the right parts of the text.

In this thesis, two evaluations will be utilized and investigated. These are the STyle EvaLuation (STEL) framework from Wegmann & Nguyen (Wegmann & Nguyen, 2021) and a method described by Zhu & Jurgens (Zhu & Jurgens, 2021) from now on called the compositionality check.

The STEL framework. The STEL framework is a modular, fine-grained, and content-controlled framework (Wegmann & Nguyen, 2021). It can test the performance of models that compare the style of two sentences. It does so by testing the model on the task shown in Figure 2. The goal of the model is to match the sentences that have the same style. The sentences are from a parallel dataset in which for every instance of a sentence having one style, there is another sentence with the same content, but a different style. To demonstrate the framework Wegmann et al. have used the formal/informal and simple/complex dimensions. Additionally, sentences where contractions (e.g. *«It is my dog»* vs *«It's my dog»*) were compared. Lastly, number substitutions were contrasted. An example would be *«Love you forever»* vs *«Love you 4ever»*.



r u a fan of them or something?

Are you one of their fans?

Oh, and also, that young physician got an unflattering haircut.

Oh yea and that young dr got a bad haircut.

Figure 2: Example of a STEL task. The anchors and sentences need to be paired according to their styles. Here, the sentences are split on formal/informal styles (Wegmann & Nguyen, 2021).

The main concern during the construction was ambiguity. This problem occurs when it is debatable which of the two parallel sentences should fall into which category. This was resolved by the use of crowdsourcing the task of classifying the sentences. Unfortunately, this will not be possible for this thesis. Another initial problem was the "Triple problem" (Figure 3). It is not known for the sentences where they are on the scale of the chosen feature. Only their relative position compared to the other sentence is known (e.g. A2 is left of A1). This would mean that if the model was asked to classify S2 without also classifying S1, it would match S2 with A1.

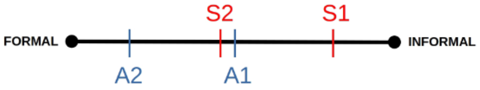


Figure 3: The triple problem in which it is shown that the sentences and anchors are relatively more or less formal than their counterparts. This would cause problems if one of the sentences was removed from the task (Wegmann & Nguyen, 2021).

2.4 *Style representation in the Dutch language*

Up until this point, almost all research has been considering the English language and its style representation. However, this thesis is focused on the evaluation of the representation of style in Dutch. Therefore, hereafter, the differences and similarities between Dutch and English style will be discussed as well as previous research into Dutch authorship verification.

2.4.1 *Style research in Dutch*

Not many authors have published research about the computational linguistic stylistics of the Dutch language (Fagel, 2009). The first major publication was by Stutterheim (Stutterheim, 1949) who described style as an individual, original use of language, which deviates from the normal use of language. It is also denoted that determining when language becomes style is yet difficult.

Building upon the publication of Stutterheim, Hellinga & Van der Merwe Scholz (Hellinga & van der Merwe Scholtz, 1955) do not see style as a deviation from the original use of language, but more as the deviation of current norms depending on the context. An important insight is that both these definitions need to have an established baseline language before uniqueness can be determined. Contrastive comparisons alleviate this oversight (Anbeek & Verhagen, 2001). Contrastive comparisons also allow for texts to be compared to each other instead of against the complete population. This is what authorship verification entails when comparing the texts of two authors.

Since a definition of style is already established in section 2.1.1 together with the fact that there should be no difference between the definitions in different languages, this definition remains the same for both languages.

2.4.2 *Stylistic differences between English and Dutch*

There is a difference in the selection and structuring of information when comparing English and Dutch (Ierland, 2010). English is a highly consistent SVO language. This means that a sentence is most often structured with a subject at the start, followed by a verb and then an object. An example would be «*Susan [S] feeds [V] the dog [O].*». In Dutch, there is no strict basic word order. While the SVO structure is possible (e.g., *Suzan [S] voert [V] de hond [O].*»), the verb typically occupies the second position in a sentence rather than directly following the subject. For example, «*Gisteren* ***ging ik*** *naar school.*» translates to «*Yesterday* ***I went*** *to school.*». This shows that in Dutch, it is possible to structure sentences differently, which can be seen as a stylistic difference.

Another example of a stylistic difference is the use of formal pronouns. Whereas English only has one singular second personal pronoun: *you*. Dutch has a formal version *u* and an informal version *jij*. Besides, Dutch also has a slightly more complex system of articles. English only uses *a/an* and *the*. In Dutch there are *een*, *de*, and *het* for which mistakes can be made.

2.4.3 *Authorship verification in Dutch*

Regarding authorship verification, there has been more research on Dutch texts. This research states that the stylometric techniques used for English AV tasks are transferable to Dutch (Luyckx & Daelemans, 2008). Besides, research suggests that a factor such as language does not affect the performance of verification models significantly when looking at English, Dutch, Greek, and Spanish (Adamovic et al., 2019; Stamatatos, 2016). This suggests that style learning techniques are transferable across languages, or at least that the representations of style are transferable.

2.4.4 *Representation of texts in Dutch*

There have only been a few attempts to create Dutch word embeddings. The attempt of Al-Rfou, Perozzi & Skiena (Al-Rfou', Perozzi, & Skiena, 2013) was one of the first but lacked the dimensionality to reach state-of-the-art performance. Then, Tulkens, Emmery & Daelemans (Tulkens, Emmery, & Daelemans, 2016) created an embedding capable of assisting models with the downstream tasks of relation evaluation and dialect identification. The results were comparable to the results achieved with the creation of Word2Vec (Mikolov et al., 2013).

Even though these embeddings are not related to style, they show that with the right data, an embedding as capable as those in English can be created using the same methodologies as in English. Examples are RobBERT (Delobelle et al., 2020) which was trained the same way RoBERTa is, but with Dutch data instead of English. BERTje (de Vries et al., 2019) was constructed in the same way but with BERT's construction method. Lastly, there is RobBERTje (Delobelle, Winters, & Berendt, 2022), which is a distilled version of RobBERT, meaning that it has increased efficiency at the cost of a small performance decline. Since these models are trained on Dutch data it would be realistic to expect their representations to have a better grasp of Dutch style than the models that were trained on English data.

3 METHODOLOGY

The methodology of this thesis consists of a few parts. These steps will lead to the evaluation of Dutch linguistic style embeddings. The first step is creating a proxy task that allows the model to learn representations that can be evaluated later. This proxy task will be a contrastive authorship verification task. With the embeddings created, the next step is to evaluate the embedding. This will be accomplished using a Dutch version of the STEL framework from Wegmann and Nguyen (2021).

This section will start with section 3.1 where I will be discussing how the data was gathered and how it was used in the CAV task. Then section 3.2 will discuss the training task and the hyperparameter tuning that was applied. Lastly, section 3.3 discusses the creation of the evaluation dataset

3.1 *Data*

The data used for this thesis originated from Reddit[3]. The data was collected and published by Cornell University[4]. It is extracted using the Convokit package. The data spans from 2015 up until 2018.

3.1.1 *Selection of the subreddits*

Reddit is a social media site comprising a collection of subreddits that each function as separate fora. This allows for a multitude of topics to be discussed on the site. Prior research has shown that this type of Reddit corpus proves to be useful for style-like models (Rivera-Soto et al., 2021). Subreddits can be about plants, psychology, or paternity leave. The versatility makes the data a good fit for this research since the goal is to create a dataset in which one author supplies contextually different utterances. Given that users can post across all of Reddit with the same username, some users post in different subreddits, which are about different topics.

Appendix 7 contains a full list of the subreddits from which data was collected for this thesis. These subreddits were selected because of the presence of Dutch comments in them. These comments can be used as utterances for the model to be trained on. The used subreddits were found in several different ways. Firstly, a list of subreddits was used[5]. This list originates from 2013 and contains outdated information, but was still valuable to select some initial subreddits. It also gave me the inspiration to look at all subreddits about provinces (e.g., r/Zeeland) and cities (e.g., r/Tilburg). I tried to find subreddits for all 12 provinces and I also looked at subreddits of the biggest 30 cities in The Netherlands. On the initial list, there was a subreddit from a Dutch University, this inspired me to look for other universities as well. The last way of finding subreddits was simply searching for the terms *Nederland* and *Netherlands* on Reddit and then selecting communities.

[3] https://www.reddit.com/

[4] https://convokit.cornell.edu/documentation/subreddit.html#dataset-details

[5] https://www.reddit.com/r/thenetherlands/comments/2bm9le/an_overview_of_the_dutch_subreddits/

Not all of these subreddits have the same number of utterances. The biggest subreddit r/thenetherlands contains 1,328,027 utterances while r/cirkeltrek is the second largest and contains 78,475. Even though there is one subreddit that is a multitude larger than the others, there is little concern that this would imbalance the dataset since the subreddit in question is full of discussions about various topics. A manual inspection points out that this is the case for all subreddits in the top five of providing utterances except r/RMTK, a subreddit focused on roleplaying the legislative branch of the government. Here, posts can still vary in their content, but they are all related to politics and most of the utterances have a similar writing style consisting of motions, questions to ministers and taking votes. All these utterances have a set format from which the authors do not deviate. This does not mean that there is no deviation between authors, it means that the differences are more nuanced so I expect that it will not jeopardize the embedding creation.

3.1.2 *Language- and other pre-processing steps*

Given that the subreddits were about The Netherlands or were aimed at Dutch users, the expectation is that the language of the collected utterances would in large part be in Dutch. However, not all subreddits have a strict language policy, meaning that there is still the possibility that there are non-Dutch utterances in the data. These utterances will be filtered out for the dataset to only contain Dutch texts. This is done by using the langdetect library[6]. The method used is as follows: langdetect is asked to determine the language of an utterance. It will return one or multiple certainty scores of the languages the utterances are estimated to be in. Since prior manual data exploration pointed out that nearly all utterances were either in English or Dutch, these are the languages that we will focus on.

The implementation is as follows: langdetect returns a list of the most probable languages. If English or Dutch is in the number one spot, this language is selected. When the first suspected language is different than these two, the second place is looked at (if it exists). If English or Dutch is in this spot, then this is selected (since neither of the two is in the first position). In the case of a top three being presented, the step for position two is repeated for position three. If neither English nor Dutch occurs, the utterance is not relevant to the dataset.

The language is filtered to only contain Dutch utterances. Since the software is probably not flawless, it is important to know how many English utterances are still in the dataset. After a manual check of 500 utterances, there were only two that were in English. Both utterances contained typical Dutch first names or place names, causing langdetect to label them as Dutch.

Apart from selecting the language of the utterance, invalid utterances were also removed[7]. Additionally, messages from bots were deleted. These bots were sought by filtering the users to see if the word *bot* was in their name. Then a manual check was done to verify if they were a bot or not. In this context, a bot is defined as a non-human user who often leaves utterances composed automatically and posted after a script is triggered.

[6] https://pypi.org/project/langdetect/

[7] These utterances are: "", " [removed] ", "[removed]", "[removed]", "[deleted]", "[deleted]"," [deleted] " inspired by (Wegmann et al., 2022)

3.1.3 *Description of data*

In total, there are 1,596,000 usable utterances after filtering. These are produced by 46,457 users. This results in a mean utterance count of approximately 34, and a median count of 3, though the mode and median are merely 1 and 3 respectively. This indicates that there are a few users with a lot of utterances and that there are many with only a few. This means that there are many users whose utterances can only be used as a mismatch since at least two utterances are required to make a matching pair.

The data originates from 92 subreddits. This means that there are approximately 17,347 utterances per subreddit. The median and mode are 205 and 1. Just as with the users, there is also no equal distribution of utterances among the subreddits. In general, the average Reddit user in the United States is more likely to be male and liberal[8]. Even though there is no specific data about the demographics of Dutch subreddits, there is data that suggests that the worldwide gender distribution of the Reddit userbase is also predominantly male[9]. This means that the model might be skewed towards males.

3.1.4 *Matching utterances*

Utterances now need to be matched to other utterances to perform the contrastive authorship verification task. Three utterances are needed per instance: The anchor, a match, and a mismatch. These can also be seen in Figure 1. A Dutch example from the dataset would be:

> Anchor: *Not sure, maar ik zou het voor het geval gewoon doen. Staat meestal ook zo in het correctievoorschift.*
>
> Match: *Ik kom uit zeeland en ik moet de laatste trein hebben, ouders kunnen me niet brengen ivm werk. De laatste trein uit zeeland gaat om 23:30 dus ik ben hoe dan ook rond 02:00/03:00 op Schiphol.*
>
> Mismatch: *Je moet er wel even rekening mee houden dat RIOT een iets andere weg in is geslagen dan "laten we de hoofdprijs zo hoog mogelijk maken". Zij betalen alle spelers van de de spring en summer competitie 15.000 dollar. Deze keuze is gemaakt in de hoop dat de lagere teams ook de middelen hebben om zich te ontwikkelen, in de plaats van een paar elite teams die ver boven de rest uitkomen.*

The anchor and the match are written by the same author while the mismatch is written by a different author. The semantic similarity between the anchor and the match must be far enough apart for the model to focus on style instead of content. Therefore, an anchor and a match are only paired if their cosine similarity score is lower than 25%. The sentence embeddings are created using RobBERT (Delobelle et al., 2020)[10] . Using these embeddings, the anchor and match can be

[8] https://www.pewresearch.org/journalism/2016/02/25/reddit-news-users-more-likely-to-be-male-young-and-digital-in-their-news-preferences/

[9] https://www.statista.com/statistics/1255182/distribution-of-users-on-reddit-worldwide-gender/

[10] https://huggingface.co/pdelobelle/robbert-v2-dutch-base

compared. After utterances occur as either an anchor, match, or mismatch, they are not reused.

A training-validation-test split is constructed with a 70-15-15 split. To prevent training data from leaking into the validation or test set, an author's utterances can only occur in one of the sets. This is achieved by setting a maximum amount of utterances to be in the initial datasets. This follows the 70-15-15 split mentioned before. Firstly, the authors are shuffled after which their utterances are added to the training data until the addition of an author leads to an excess of the allowed utterances. In this case, all the author's utterances go to the validation set, which in turn overflows into the test set. This ensures a random distribution of the authors according to the desired distribution rates. This results in a training set with a size of 350,000 triplet pairings and a validation and test set size of 75,000 triplet pairings. As a comparison: Wegmann et al. (2022) had 210,000 triplet pairings for their training test and 45,000 for their validation and test sets. Additionally, my training set has 26,839 authors that contribute a text versus the 270,079 from Wegmann et al. The validation set has 7,263 authors (versus 57,352) and the test set has 6,420 authors (versus 57,762).

3.1.5 *Limiting author texts*

Researchers from the Blablablab[11] have found that limiting the number of texts an author can contribute to the training, validation, and test sets improves the performance of the model. The idea is that including too many texts from just a few authors can prevent the model from learning a sufficiently diverse range of writing styles. It could thus be beneficial if the model has access to a more diverse array of styles.

To see if the created dataset suffers from an over-representation of certain authors, the texts of authors were counted. The top ten contributors per data set are displayed in Table 1. As can be seen here, the top contributor makes up 1.2% of all the sentences in the training set (respectively 2.9% and 3.7% for the validation and test set). To prevent these authors' styles from playing too big of a role in the model's fine-tuning process, a new dataset was created where a maximum is set to the number of texts an author can contribute. For the training set, this maximum is set to 2,500. For both the validation and test sets, the max is 1,000.

Using these limits, the datasets are constructed in the same manner as the previously created dataset. Due to the limit that is applied, there were not enough matches found for the validation and test set. This means that the size of the validation set shrunk to 70,000. The test set was reduced to 67,500 task instances. The training task's number of task instances remained at 350,000.

This change allowed more authors to be in the datasets. In the training task, the number of authors increased to 27,863 (an increase of 1,024). The validation set has 6,747 authors (a decrease of 16) and the test set has 7,214 authors (an increase of 794). Although there is a small decrease in the number of authors in the validation set, at the same time the size of the dataset also shrunk by 7,500 instances.

[11] https://blablablab.si.umich.edu/

Both of these datasets will be tested in this thesis to see if limiting the number of texts per author and thus hopefully introducing more stylistic variability to the model helps with the performance of the model. The

Ranking	Training set	Dev set	Test set
1	12,217	6,537	8,420
2	11,252	5,636	5,887
3	8,053	4,693	4,502
4	7,898	4,301	4,325
5	7,818	3,909	3,413
6	7,385	2,738	2,508
7	6,789	2,734	2,304
8	6,691	2,677	2,111
9	6,310	2,477	2,040
10	6,039	1,877	1,822

Table 1: Number of texts that the most frequently appearing authors contribute to their respective dataset. No author can appear in more than one set.

3.2 *Training task: Contrastive authorship verification*

Contrastive authorship verification is used as a training task to force the model to learn a representation of style. By forcing the model to match two semantically different sentences from the same author and making the other choice a random text from another user, the model has to find attributes other than content to distinguish authors. This proxy task can be divided into smaller segments. The first step, the data preparation has already been completed. After this, the next step is to select a model that is suitable for fine-tuning.

As discussed in section 2.4.4, several BERT-like models, such as RobBERT (Delobelle et al., 2020) are trained to create Dutch content representations. This means that they possess knowledge of the Dutch language, which is more useful than knowledge of English. The base model already has representations of the content of Dutch texts. These initial representations are used as a base. The embeddings of the base model are adjusted using my training task training task. There are 3 candidates to use as a baseline. These are: BERTje (de Vries et al., 2019), RobBERT (Delobelle et al., 2020) and RobBERTje (Delobelle et al., 2022). In similar research for creating English-style representations, RoBERTa was found to achieve better results than BERT (Wegmann et al., 2022). Considering that RobBERT was created similarly to RoBERTa, I chose to use this model as a base. It also has the benefit of a slight performative advantage over RobBERTje.

3.2.1 *Use of seeds*

Seeds ensure that produced results are replicable. The finetuning process is stochastic, meaning that there is randomness in the training process. By setting a seed, the randomness is removed and the training process becomes non-stochastic, meaning that others can run the same code without receiving different results.

Running the model on 3 different seeds allows us to generalize the model and not judge the model on the result of only 1 seed. Therefore, running the model with several seeds will give a more reliable view of the results. On top of that, Dutta, Arunachalam, and Misailovic (2022) give two recommendations: The first is to only use exact seeds when in need of exact reproducibility. The second is to use a random seed that is logged. This ensures that the reproducibility is not compromised while the code exhibits different sequences. Considering these recommendations, three seeds [8, 1409, 1812] were chosen and used across all fine-tuning tasks. Since exact replicability is crucial for others to reproduce my code and randomness adds no value to the seed selection process, choosing these three seeds seems like a suitable approach.

3.2.2 *Loss- and evaluation function margin value*

Taken from Wegmann et al. (2022), the chosen loss function for the CAV is the triplet loss (Schroff, Kalenichenko, & Philbin, 2015). This loss function holds the form of $L = max(d(a,p) - d(a,n) + m, 0)$. Here the distance d is calculated for both the anchor and positive sample (a, p) and the anchor and negative sample (a, n). Then, a margin m is added that decides how far the model needs to keep the anchors and mismatches apart. This method ensures that positive and negative samples are separated while ensuring that they are still grouped with other similar samples. The margin is the only hyperparameter of which multiple values are tested in this thesis. It tells the loss function how far away positive and negative samples should be from each other.

Parameter	Value
Loss	[triplet, contrastive]
Evaluator	triplet
Margin	[0.4, 0.5, 0.6]
learning rate	2-e4
eps	1-e8
warmup steps	10% of training
batch size	8
epochs	4
eval batch size	4

Table 2: Training parameters

3.2.3 *Other hyperparameters*

The other hyperparameters were taken from prior research into finetuning a BERT-like model for the English language (Wegmann et al., 2022) and can be found in Table 2.

The triplet evaluator verifies whether the model has minimized the distance between anchors and matches while maximizing the distance between anchors and mismatches. This is done by seeing if the matches are closer to the anchor than the mismatches are.

3.3 *Creation of a Dutch evaluation dataset*

In line with Research question 2, the next step of the research is to create a dataset that contains sentences that have the same content, but that differ in one specific aspect of style. In English, this dataset already exists (Wegmann & Nguyen, 2021). The STEL framework consists of pairs of sentences that differ only on certain stylistic features, but that do hold the same semantic meaning. In this subsection, all subtasks of the STEL dataset are discussed in detail. Examples of all rewritten sentences can be found in Table 3

This research builds upon previous work of student assistants at Utrecht University. They have developed simple/complex and contraction dimensions of the Dutch STEL variant. The other three dimensions will be developed by me.

char / dimension	Original sentence	Adapted sentence
Contraction	Wat had 'ie precies fout..? En valt dom, vergooit daarmee z'n kansen op een medaille.	Wat had hij precies fout..? En valt dom, vergooit daarmee zijn kansen op een medaille.
Punctuation	Het RIVM meldt slechts 1 nieuwe overleden corona patient. Op naar de nul!!! Het werd ook hoogtijd,dat mensen zich meer en meer op de werkelijke belangrijke zaak gaan richten: NATUURBELANGEN!!! Wanneer de natuur kapot zal zijn, gaat ook de mensheid kapot! Zo is het en niet anders!!!! url	Het RIVM meldt slechts 1 nieuwe overleden corona patient. Op naar de nul. Het werd ook hoogtijd,dat mensen zich meer en meer op de werkelijke belangrijke zaak gaan richten: NATUURBELANGEN. Wanneer de natuur kapot zal zijn, gaat ook de mensheid kapot. Zo is het en niet anders. url
Capitalization	IS DIT NOU ECHT WAT WIJ WILLEN: WINKEL VERZEGELEN ? HAD DIT NIET VERSTANDIGER GEKUND ? MOET ER PERSÉ EEN "VOORBEELD" GESTELD WORDEN ? Meute jouwt politie uit na verzegelen 'corona-supermarkt' {[}url{]} via telegraaf SHOCKING!!! Mensen, grafiek laat zien TWEEDE LOCKDOWN ABSOLUUT NOODZAKELIJK!!!!!!!!!!!!!!! {[}url{]}	Is dit nou echt wat wij willen: winkel verzegelen ? Had dit niet verstandiger gekund ? Moet er persé een "voorbeeld" gesteld worden ? Meute jouwt politie uit na verzegelen 'corona-supermarkt' {[}url{]} via telegraaf Shocking!!! Mensen, grafiek laat zien tweede lockdown absoluut noodzakelijk!!!!!!!!!!!!!!! {[}url{]}
Simple / Complex	Hij stierf in Freiburg im Breisgau aan de pest in 1538 op terugreis naar de Nederlanden. Mogelijk gaat het hier om latere toevoegingen aan het gedicht.	In 1538 op terugreis naar de Nederlanden stierf hij in Freiburg im Breisgau aan de pest. Potentieel gaat het hier om latere toevoegingen aan het gedicht.
Formal / Informal	Wel grappig,.. hoe hebben ze die Cruiseschip in Italië zo snel weten te testen? Corona is zo slim dat het savonds pas actief wordt?	Wel grappig; hoe hebben ze iedereen op dat cruiseschip in Italië zo snel getest? Blijkbaar is corona zo slim dat het 's avonds pas actief wordt?

Table 3: Rewritten sentences for the STEL framework. The sentences on the left represent the sourced sentences. For the simple/complex sentences, the sentences on the left are the simple sentences and the ones on the right are the complex sentences.

3.3.1 *Overview of the evaluation dataset*

The Dutch evaluation dataset consists of characteristics and dimensions. The characteristics are small changes to sentences. In this thesis, they are capitalization, punctuation, and contraction usage. The dimensions involve more significant alterations to sentences, often including modifications to their structure. Here, the used dimensions are simple/complex and formal/informal.

3.3.2 *General problems with the creation of a dataset*

The problem is that style is not directly transferable between languages. So, rules for the English language might not be applicable to Dutch. In English, there are different types of contractions when compared to Dutch for example. This lack of direct applicability is particularly evident in the case of style dimensions. Here there are no clear quantifiable aspects to determine if a text belongs to one category or the other. For example in the simple/complex dimension. There are options to simplify a text in English (Feng, 2008; Xu, Napoles, Pavlick, Chen, & Callison-Burch, 2016). These are splitting, deletion, and paraphrasing. Since these are basic techniques, it can be assumed that these rules are also relevant to Dutch. The problem arises when deciding how to apply the techniques. In English, research has been conducted into the specifics (Harley, 2013). In Dutch, other rules have been established (Temnikova, 2012).

3.3.3 *Gathering and processing the data*

For the creation of the Dutch equivalent of this dataset, three sources were used. The first is Twitter data. This dataset was created in 2020 using the Twitter API and published as the Dutch Social media collection on Kaggle[12]. For this thesis, only four out of ten available chunks were used. This means that there are a total of 108,428 tweets in the dataset. This data was gathered by selecting tweets that were either in Dutch and/or for users that had indicated that they were within the Dutch geographical region.

To this data, some adjustments were made. The mentions (@'s) and hashtags (#'s) were removed from the utterances e.g. *@NOS* becomes *NOS*. Additionally, all messages that are retweets start with *RT @[username]:*. These were also removed since they were not of the user's doing. An additional language check was also conducted to ensure that all utterances were in Dutch. The original dataset had also sampled Tweets sent in the geographic areas where Dutch is the main language causing some other languages to enter the dataset. Lastly, all URLs were replaced with [URL] to match this to the data that was used for the CAV task. This data was used to create datasets for punctuation, capitalization, and the formal/informal dimension.

The second source is Dutch Wikipedia from which data was gathered previously by students assistants from Utrecht University. Wikipedia lists ten base categories[13]. These all contain subcategories which in turn also contain subcategories. By doing a depth-first search, texts from a wide variety of topics can be gathered. It also ensures that there is no over-abundance of subcategories that are over-represented. 2,000 sentences were sampled from each base category, resulting in 20,000 texts. A part of these sentences were used for the simple/complex dimension.

The third and last source is Reddit. The student assistants from Utrecht University gathered data from nine subreddits[14] between 2015 and 2018. To this data, a breadth-first approach is applied using regular expressions to find texts that are likely to contain contractions. Using this method, 3,923 texts were collected for the contraction task.

[12] https://www.kaggle.com/datasets/skylord/dutch-tweets?resource=download&select=dutch_tweets

[13] https://nl.wikipedia.org/wiki/Categorie:Alles

[14] The data originates from https://www.reddit.com/r/thenetherlands/comments/2bm9le/an_overview_of_the_dutch_subreddits/ it is not stated from which nine subreddits data was sampled, only that the subreddits were manually inspected to ensure enough Dutch sentences were present in the subreddits.

3.3.4 *Capitalization*

When a sentence is completely capitalized, it propagates a certain intensity that a standardly capitalized equivalent would not. *«I LOVE FROGS.»* suggests that the author of the text truly adores frogs whereas the author of *«I love frogs.»* appears to have a less profound relationship with frogs.

For this capitalization task, I extracted sentences from the Twitter dataset where at least 50% of the characters were capitalized. These sentences were then manually checked to see if they were useful enough to be included in the dataset. This meant that interjections e.g. *«HAHAHAHA»* were removed, just like texts that only contained a few words. Thereafter the sentences were manually transformed to standard capitalization since no tools reliably do this in Dutch.

Another option would have been to decapitalize all sentences and only use lower-case letters. I decided not to do so and instead transformed the sentences to standard Dutch because this would normalize the sentences more. If the sentences had been all lowercase, it would mean that all names, place names, and other words that should be capitalized would be capitalized wrongly. This would create a difference between sentences with many words that should be capitalized and sentences that contain none to a few of those. In principle, this does not have to be a big problem, but given the choice, the current method seems to be a better fit.

3.3.5 *Contraction*

The data for this task has been gathered by Student assistants at Utrecht University and is inspired by Wegmann and Nguyen (2021) who also include contractions in their STEL task. Contractions are shortenings of words that retain their original meaning. The use of contractions is seen as informal in both English and Dutch. Though the usage of contractions is generally considered correct in Dutch spelling[15]. The choice to use contractions could stem from laziness or a wish from an author to make their text more compact. On the other hand, it could also be an unconscious decision from the author. Although the style of a text changes when contractions are written as fully spelled-out words, texts with fully written words convey a distinct tone.

For the contraction dataset, student assistants from Utrecht university sought two types of contractions. The first is, as is it in English, made up of apostrophes that merge words or parts of words e.g. *«Zijn»* (His) becomes *«Z'n»*. The second type are words that can be shortened without the use of an apostrophe e.g. *«Ik ben niet»* (I am not) becomes *«Ik bennie»*. A list was made to detect these in texts. The downside with this method is that the instances where the contractions are written without the apostrophe are not detected (*«Zn»*). A full list of all contractions can be found in Table 4.

From the gathered data, 500 sentences were manually rewritten by a native Dutch annotator. All contractions were changed to their non-contracted variant. The sentences were also filtered.

[15] https://onzetaal.nl/taalloket/me-zusje-mn-zusje-mijn-zusje

Code	contraction	normal	Code	contraction	normal
mn	m'n	mijn	s	's	eens
zn	z'n	zijn	hebbie	hebbie	heb jij
t	't	het	kleurn	kleur'n	kleuren
m	'm	hem	matn	mat'n	maten
ie	'ie	hij	soortn	soort'n	soorten
dr2	d'r	er	zon	zo'n	zo een
n	'n	een	niewaar	niewaar	niet waar
dr1	d'r	haar	dikdakn	dikdak'n	dikdakken
em	'em	hem	boonn	boon'n	boonen
mneer	m'neer	meneer	mvrouw	m'vrouw	mevrouw
naja	naja	nou ja	fluitnt	fluitn't	fluitent
k	'k	ik	tuurlijk	tuurlijk	natuurlijk
adam	A'dam	Amsterdam	drbij	d'rbij	erbij
tis	tis	het is	kweenie	kweenie	ik weet niet
ff	ff	even	googln	Googl'n	Googelen
hijs	hijs	hij is	m2	m	hem
noordn	noord'n	noorden	hm	h'm	hem
welvarent	welvaren't	welvarend			

Table 4: Pairs of contracted words and their correct Dutch equivalent accompanied by their code.

3.3.6 *Punctuation*

How an author uses exclamation marks can say a lot about the style of an author. This goes further than its primary linguistic function of expressing surprises or conveying shock or a strong emotion[16]. In addition, an exclamation mark can play a role in sentiment analysis where the perception of a review can be influenced by the use of the punctuation (Teh, Rayson, Pak, & Piao, 2015). Using more exclamation marks amplifies the perception of positivity in a positive review, while in a negative review, it intensifies the perception of negativity. Additionally, using exclamation marks is associated with friendliness (Nicoladis, Duggal, & Besoi Setzer, 2023). It does need to be noted that this was only tested on positive messages. About the effect exclamation marks have on negative texts Taboada, Trnavac, and Goddard (2017) only briefly mention that their use may contribute to conveying more negativity. Lastly, excessive use of an exclamation mark indicates a more intense opinion than the use of only a singular one. E.g. «*I love dogs!*» is less intense than «*I love dogs!!!!!*».

Considering this, we want to find sentences with excessive exclamation mark usage. This is done by scouring the Twitter data and filtering texts that contain at least three consecutive exclamation marks («*!!!*»). The filtered texts were then manually checked to see if their transformation had been successful. Additionally, it was checked if their language was Dutch and if the sentences contained enough information to be used in the dataset. This means that sentences containing only one word or interjection (a standalone word or phrase expressing spontaneous emotion or reaction) were removed.

[16] https://dictionary.cambridge.org/grammar/british-grammar/exclamations

The inclusion of other punctuation marks would have been a possibility. Question marks and ellipses were also considered, but ultimately not implemented. The use of a question mark, if used solitarily, does not necessarily change the style of a sentence; «*You have a dog.*» and «*You have a dog?*» do not differ greatly in perceived style. Also, the inclusion of a question mark is not a stylistic choice, but a necessity if the message needs to be conveyed correctly. A substitution of a question mark does not have the same effect as that of an exclamation mark. Lastly, an option would be to replace excessive question mark usage «*You have a dog???*» with only a singular question mark. The problem with including this in the same task as the exclamation marks would be that the substitutions of these two punctuation marks do not constitute the same shift in style. The removal of punctuation marks indicates a different shift than that of question marks.

Ellipses can also be used to alter the style of a sentence and could therefore be considered to be included in this dataset. In computer-mediated communication (CMC) ellipses are used to indicate sarcasm (Vandergriff, 2013) meaning that removing them not only changes the style of the text, but also the meaning. E.g. «*Love you too...*» has a completely different meaning than «*Love you too.*». Therefore, ellipses are not included in this dataset.

3.3.7 *Simple / complex*

The simple complex frame was created by student assistants from Utrecht University, inspired by Wegmann et al. (2022) who also use simple/complex as a style dimension in their research.

The texts were rewritten based on two guides on Dutch websites advising on how to make texts more accessible to those with a lower literacy level[17] combined with sources on text complexity (Temnikova, 2012). Out of the elements that make up text complexity found in Temnikova (2012)'s research, lexical and syntactical complexity form the basis of the guidelines given on the sites. Lexical complexity refers to the richness and sophistication of vocabulary used in spoken or written language. Advice given to alleviate this complexity for readers is not to use loan words, homonyms, jargon, or abbreviations. In contrast, syntactical complexity concerns a sentence's structure in terms of grammar, syntax, and the relationships among its components. Tips to help with these difficulties consist of splitting sentences into two separate ones, and not using figurative language.

To minimize the lexical complexion, difficult words were exchanged for easy words. Since commonly used words are more likely to be understood by readers when compared to less frequently used words (Temnikova, 2012). Thus, the Dutch SoNaR corpus (Oostdijk, Reynaert, Hoste, & Schuurman, 2013) was used to determine the frequency of word occurrences in the Dutch language. The less frequent words were, where possible, exchanged for their more often occurring counterparts.

To decrease syntactical complexity, there are five rules of which examples are given in Table 5. The first rule is to have the sentence follow the standard word order. This word order means that the sentence starts with the subject, followed

[17] https://www.stichtingmakkelijklezen.nl/kenmerken-makkelijk-lezen & https://www.lezenenschrijven.nl/wat-doen-wij/oplossing-voor-je-vraagstuk/kennisblad-eenvoudige-taal-voor-laaggeletterden

by the verb, and then the object (SVO). These sentences are thought to be easier to process (Greer, Rice, & Deshler, 2014). The reason for this is that SVO is the most common sentence structure in Dutch (Harley, 2013).

The second rule is to write sentences in an active voice. These sentences are easier to process (Harley, 2013). This is based on Chomsky's theory of transformational grammar (Chomsky, 2002). This states that all sentences can be reduced to kernel sentences that are always written in the active form. To these base sentences, adjustments can be made to change the sentences. One of these is passivation.

The third rule is to keep verbs close together and close to the subjects. This is based on the fact that readers forget word order very quickly (Harley, 2013). If the verbs are close together, the whole action becomes clear immediately to the reader e.g. in the sentence *«De race werd in Japan gereden in Japan.»* (The race was raced in Japan). The verbs *werd* and *gereden* are apart. When the sentence is changed to *«De race werd gereden in Japan.»* (same meaning), the verbs are close together, meaning that less computation is needed to comprehend the meaning of the sentence. In this example the subject *race* is also immediately followed by the verbs, making clear what effect the verbs have on the subject.

The fourth rule is to not use embedded clauses. These embedded clauses, *like these words are an example of*, are hard to process (Harley, 2013). Not only must the main sentence be understood, but now the additional information must also be processed in the middle of processing the main sentence. When this subclause is moved to the end of the sentence, it allows for information to be processed linearly instead of in parallel, which makes it easier.

The last rule is to avoid joined sentences and split them into two separate sentences. This rule is based on Harley (2013)'s statement which states that working memory is only processed at the end of a sentence (Jarvella, 1971). This means that processing multiple sentences is easier than processing one long sentence because the brain has time to process the contents of the sentences one by one.

3.3.8 *Formal/Informal*

The creation of the formal/informal dimension is inspired by the work of Pavlick and Tetreault (2016). The researchers concluded that humans have a coherent sense of the concepts of formality. To better understand what made a sentence informal or formal, they chose 1,000 informal sentences that were rewritten into formal sentences. After this, they manually reviewed 100 sentences to see which changes had been applied. These findings were taken as a rough guideline to transfer sentences from informal to formal versions.

These informal sentences are from the Twitter dataset and were manually picked to make sure that the sentences were indeed informal. These sentences were then rewritten to standard Dutch sentences using my knowledge of the language, which is my mother tongue. The sentences were then checked on their grammatical correctness by another native Dutch speaker.

Rule	Simple sentence	Complex sentence
Use standard word order (SVO)	De finish ligt ook op die plek in 2019. The finish will also be at that location in 2019.	Ook in 2019 ligt de finish op die plek. Also in 2019, the finish will be at that location.
Lexical complexity	Het verschil tussen aardwetenschappelijke termen en begrippen is te groot. The difference in earth science terms and concepts is too great.	De verscheidenheid aan aardwetenschappelijke termen en begrippen is te groot The variety of earth science terms and concepts is too great
Write Active Sentences	Devin Beats mixte en masterde het nummer. Devin Beats mixed and mastered the song.	Het nummer werd gemixten gemasterd door Devin Beats. The song was mixed and mastered by Devin Beats.
Keep Verbs Close Together and Close to the Subjects	De races werden verreden op 28 augustus1988 op het Sportsland SUGO nabij Murata, Japan. The races were held at the Sportsland SUGO near Murata, Japan on August 28, 1988.	De races werden op het Sportsland SUGO nabij Murata, Japan verreden op 28 augustus 1988. The races were at the Sportsland SUGO near Murata, Japan held on August 28, 1988.
Do Not Use Embedded Clauses	Oorspronkelijk, in de 15e eeuw, werd met 'Dune du Pyla' een zandbank bedoeld die iets ten noorden lag van de huidige duin. Originally, in the 15th century, 'Dune du Pyla' referred to a sandbank located slightly north of the current dune.	Oorspronkelijk, in de 15e eeuw, werd met 'Dune du Pyla' een zandbank, die iets ten noorden lag van de huidige duin, bedoeld. Originally, in the 15th century, 'Dune du Pyla', located slightly north of the current dune, referred to a sandbank.
Avoid joined sentences	Hij ontwierp ook het personage Angol Fear voor het spel Soulcalibur IV. Hij is conceptontwerper voor de Kemono Friends franchise. He also designed the character Angol Fear for the game Soulcalibur IV. He is a concept designer for the Kemono Friends franchise.	Hij ontwierp ook het personage Angol Fear voor het spel Soulcalibur IV en is conceptontwerper voor de Kemono Friends franchise. He also designed the character Angol Fear for the game Soulcalibur IV and is a concept designer for the Kemono Friends franchise.

Table 5: Rules to make sentences more simple.

4 RESULTS

In this chapter, two types of scores will be discussed. The first ones are the scores on the contrastive authorship verification task (CAV), where the goal of the model is to match texts that are from the same author. The second task is the STEL task in which the model is evaluated on its ability to match texts based on their style. The ultimate goal of this thesis is to see how well the model can represent style. Therefore, the most important metric is the score of the model on the STEL task.

4.1 *CAV task*

In the CAV task, the goal of the model is to match anchor A to one of two other sentences that are written by the same author. One of the sentences is semantically distinct from the anchor while having the same author. The other sentence is from another author. There are validation scores and test scores. The cosine similarity is used to compare two texts on their closeness in the representation. This was used for the validation scores and it shows how well the model was able to learn while the test scores are the accuracy scores on the test set using the triplet evaluator.

4.1.1 *Validation scores*

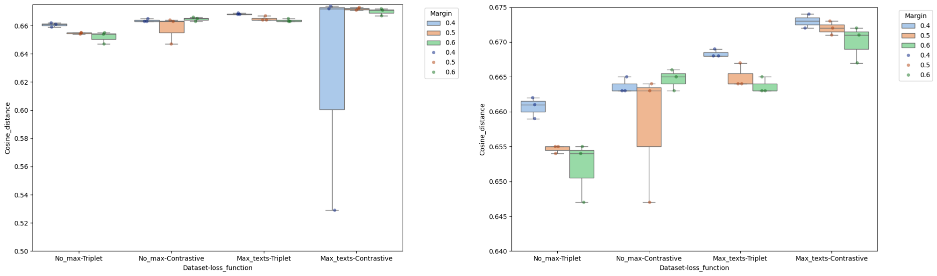
The cosine similarity score can be found in Table 6. There is no baseline for the validation score since the base model is not fine-tuned on the data. Fine-tuning RobBERT with a contrastive loss and a margin of 0.4 on a dataset that sets a maximum on the number of texts an author can contribute to the dataset yields the best results on the validation set. It can also be seen that in general, scores go down as the margin goes up. The only exception is when using a contrastive loss on the dataset which sets no limits to the number of texts per author. Additionally, using contrastive losses always leads to a better score, as does using a dataset where the number of texts per author is limited. A visualization of this can also be seen in Figure 4b. It should also be noted that all scores on these tests are close together and range from 65.2% to 67%.

4.1.2 *Removal of non-learning seeds*

When looking at the validation scores of the models, it becomes clear that for some of the trained models there are significant differences between the performance of different seeds. This is exemplified by the high standard deviation scores seen in Table 6. Additionally, it is the case that almost all of the scores for the models on the STEL task are exactly 0.5, indicating that there is no training happening in these instances and that the model is guessing. Some of the models were retrained as has been done in Wegmann et al. (2022). Due to time restraints, not all were. Instead, failing seeds were removed from the data. The results from this can be seen in Figure 12. For the remainder of the analyses, the scores originate only from learning seeds. This means that the scores of 2 seeds are unused: for both datasets there is one seed missing from the models trained with contrastive loss and a margin of 0.4.

Dataset	Loss	Margin	Dev set cosine similarity	Test score accuracy
No max	RobBERT		-	0,526
	T	0.4	0,661(0,001)	0,662(0,002)
		0.5	0,655(0)	0,656(0,001)
		0.6	0,652(0,004)	0,656(0,001)
	C	0.4	0,664(0,001)	0,668(0)
		0.5	0,658(0,008)	0,665(0,002)
		0.6	0,665(0,001)	0,665(0)
Max	RobBERT		-	0,522
	T	0.4	0,668(0)	0,665(0,001)
		0.5	0,665(0,001)	0,665(0,001)
		0.6	0,664(0,001)	0,659(0,001)
	C	0.4	**0,673(0)**	0,669(0)
		0.5	0,672(0,001)	**0,67(0,001)**
		0.6	0,67(0,002)	0,668(0,003)

Table 6: **CAV dev and test Results.** The results of six different fine-tuned RobBERT models are shown. Three types of accuracy are shown for the results of the CAV task on the test data, as well as the standard deviation. The best performance per column is **boldfaced**. There is no baseline for the dev set since this is measured using validation data during training. The data reflects the mean and standard deviation of three seeds: [8, 1419, 1812].



(a) Cosine distances for all seeds.

(b) Cosine distances only for succeeding seeds.

Figure 4: Boxplots combined with stripplots illustrate the cosine scores obtained on the validation data for the CAV task. The plots are grouped based on whether the dataset had no author restrictions and whether contrastive or triplet loss was employed. Margin values are distinguished using different colors.

4.1.3 *Test set scores*

The test scores are calculated by the triplet evaluation function and can be found in Table 6. Fine-tuning RobBERT with a contrastive loss and a margin of 0.5 on a dataset that restricts an author's texts yields the best results on the validation set. However, the performance of the validation data decreased as the margin increased, and this is not exactly the case for the test data. However, it is still the case that using a margin of 0.4 results in the best scores, except for the contrastive loss with the limiting dataset. On top of that, the contrastive loss also outperforms the triplet loss and the use of a limiting dataset still proves useful though the

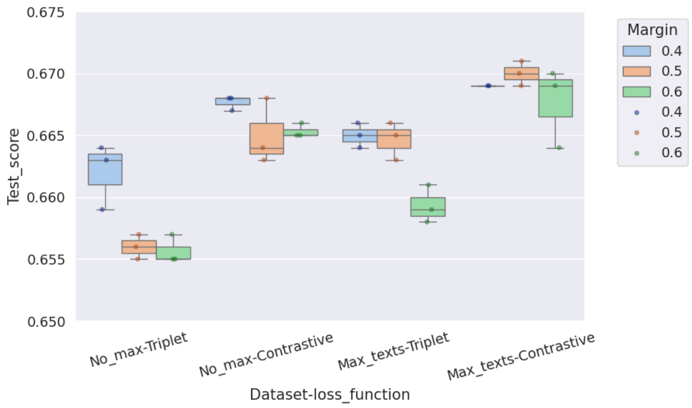


Figure 5: Accuracy of fine-tuned models on the test set. Categorized per used dataset and used loss function. The margins are represented by their own respective color.

differences are smaller than they were in the validation set. The baseline is the score that RobBERT achieves on the test set. As can also be seen in Figure 5, all fine-tuned models outperform the base model for both datasets.

When comparing this score to the results of Wegmann et al. (2022), their best scoring model is a fine-tuned RoBERTa model with an accuracy score of 0.77. My best scoring model reaches a score of 0.67. It should be noted that RoBERTa-base, achieves a score of .63 compared to RobBERT's score of 0.53. This means that fine-tuning boosts the performance of both models by 14 percentage points. It should be taken into consideration that these tasks were performed on different datasets using similar, but ultimately different techniques and that a comparison between the two is not as easily made as it might seem.

4.2 *STELNL scores*

For the STEL task, there are two types of scores: there is the original STEL task in which four sentences are given to the model with the task of matching the sentences that have the same styles. Then, there is also the STEL-or-content task in which only three sentences are presented. Here the model is tasked with matching the two sentences with a matching style. The STEL-or-content task is constructed to test if representations prefer style over content. An overview of the STEL scores can be found in Table 7. An example of a task would be the following contraction task:

Anchor 1: *Ik ken 'm nauwelijks!*

Anchor 2: *Ik ken hem nauwelijks!*

Sentence 1: *Mijn eigen kei-week was vijf jaar geleden, en ik heb nog steeds contact met mensen uit m'n groep!*

Sentence 2: *Mijn eigen kei-week was vijf jaar geleden, en ik heb nog steeds contact met mensen uit mijn groep!*

For this example, the quadruple setup would require the model to pair anchor 1 to sentence 1. The triplet task would remove anchor 2 from the task and it would then prompt the model to pair anchor 1 with the corresponding sentence (1 in this example)

For all STEL tasks, the base model outperforms the fine-tuned model in the original task. In this task, four sentences are presented, and the style pairs must be matched by the model. For the STEL-or-content task, the opposite is true. Here, all the fine-tuned models outperform the base model, except for the capitalization task.

Dataset	Loss	Margin	Capital (N=77)		Cont′ (N=3990)		Punct. (N=244)		Simple (N=8520)		Formal (N=50)	
			o	o-c	o	o-c	o	o-c	o	o-c	o	o-c
	RobBERT		**.99**	**0.53**	**.97**	0.01	**.98**	0	**.53**	0	**.80**	0
No max	T	0.4	.89±.02	.32±.06	.95±.01	.08±.01	.96±0	.05±.01	.51±.00	.02±00	.71±.01	.19±.01
		0.5	.87±.02	.29±.01	.94±.02	.09±.03	.93±.02	.05±.01	.51±.00	**.03±.00**	.69±.07	.22±.03
		0.6	.85±.02	.43±.09	.94±.01	**.09±.01**	.94±.01	**.07±.02**	.51±.00	**.03±.00**	.66±0	**.25±.03**
	C	0.4	.92±.03	.32±.03	.96±.00	.06±.01	.95±.01	.02±.01	.51±.01	.01±.00	69±.07	.17±.02
		0.5	.93±.01	.31±.04	.97±.00	.06±.02	.96±.01	.02±.01	.51±.01	.01±.00	.65±.01	.12±.00
		0.6	.92±.02	.32±.07	.96±.00	.06±.02	.94±.01	.01±.00	.51±.00	.01±.00	.70±.02	.14±.02
Max	T	0.4	.83±.06	.26±.11	.94±.00	.04±.00	.94±.02	.04±.00	.51±.00	.02±.00	.69±.05	.17±.02
		0.5	.84±.06	.29±.08	.94±.00	.04±.01	.94±.00	.04±.02	.51±.00	.02±.00	.65±.02	.21±.02
		0.6	.83±.05	.26±.01	.92±.01	.06±.01	.92±.00	**.07±.02**	.52±.00	**.03±.00**	.64±.04	.23±.04
	C	0.4	.90±.03	.20±.03	.96±.00	.04±.01	.97±.01	.02±.00	.51±.0,00	.01±.00	.65±.01	.15±.01
		0.5	.86±.03	.21±.03	.96±.01	.02±.00	.95±.01	.01±.00	.51±.00	.01±.00	.71±.04	.16±.00
		0.6	.89±.04	.20±.02	.95±.00	.03±.00	.95±.00	.02±.01	.51±.01	.01±.00	.67±.01	.16±.00

Table 7: Accuracy per instance of the STELNL task. The average is measured by taking all correct predictions divided by the total number of instances.

In the next part, the STEL tasks will be discussed one by one since there are nuances per task that should be highlighted separately.

4.2.1 *Capitalization*

Regarding the capitalization task, it is interesting to note that this is the only task in which the base model can outperform the fine-tuned models in both the original and the STEL-or-content tasks. In general, there are no patterns to be discovered in the margin values for either task. For the original task, the use of contrastive loss helps the model score better when compared to the use of triplet loss. The same cannot be said for the STEL-or-content task. It also seems like the use of a restrictive dataset leads to worse performances for both tasks.

4.2.2 *Contraction*

When it comes to the contraction task of the STEL framework, it seems to be the case that increasing the margin leads to worse results and that using contrastive loss leads to better results. Using a restrictive dataset does seem to make results slightly worse. For the STEL-or-content task, the margin has no effect on the performance of the score, but the use of triplet loss proves to be more beneficial than the use of contrastive loss. Finally, the author limitations decrease the performance of the models.

It should be noted that this is the only STEL task where there is a fine-tuned seed that outperforms the base model in the original task.

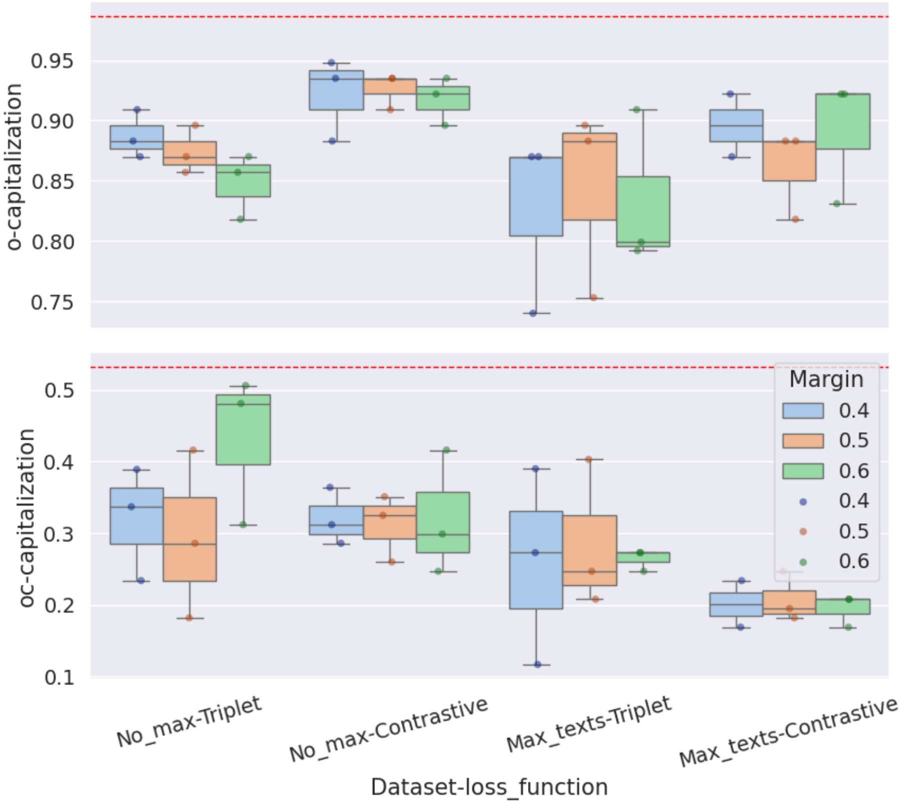


Figure 6: Boxplots and stripplots of the scores the fine-tuned models on the STEL capitalization tasks. The X-axis represents the dataset used to train on and the used loss function. The red dotted line is the performance of the base model.

4.2.3 *Punctuation*

As was the case for the earlier mentioned tests, the punctuation test does not have a clear benefit of changing the margin used. Instead, it appears that there is no direct correlation between the margin value and the score on the original STEL task. Using contrastive loss generally increases the score. Setting a limit on the texts an author can contribute, boosts the score of the model. All fine-tuned models perform worse than the base model.

The opposite is true for the STEL-or-content task. Here all fine-tuned models perform better than the base model. Using a triplet loss works better than using a contrastive one. When using a triplet loss, an increase in margin means an increase in performance. This is not the case for contrastive loss. It does not seem to make a great difference which dataset is used.

4.2.4 *Formality*

For the original STEL task, the base model outperformed the other models on the formality task. Apart from this, it only seems like the margin plays a role

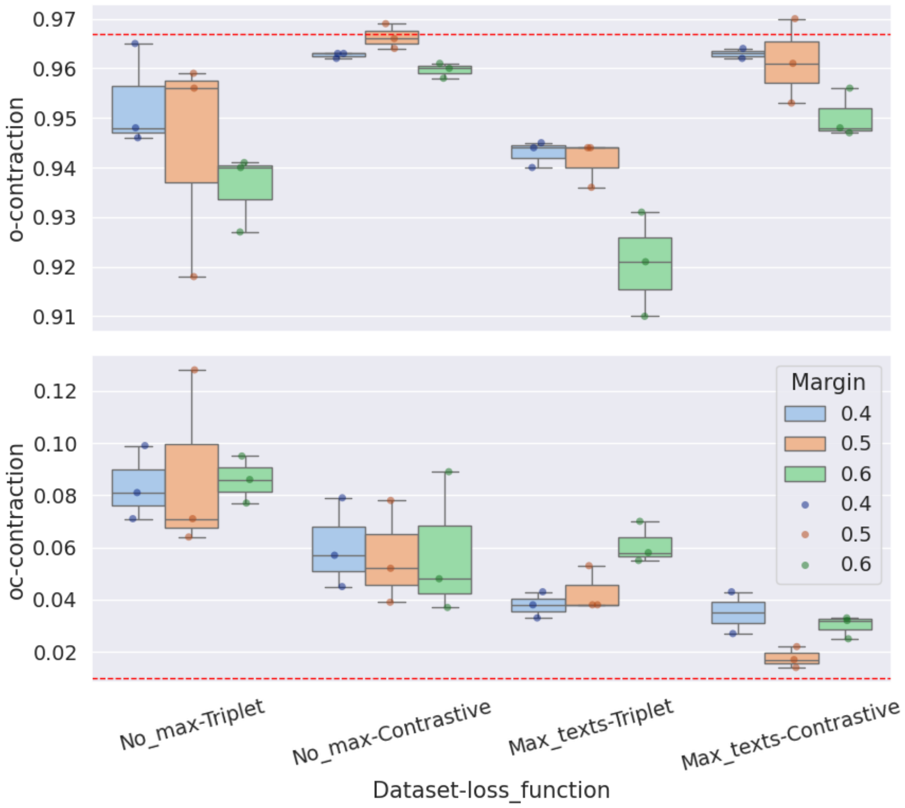


Figure 7: Boxplots and stripplots of the scores the fine-tuned models on the STEL contraction tasks. The X-axis represents the dataset used to train on and the used loss function. The red dotted line is the performance of the base model.

when using the triplet evaluator. This is the same in the STEL-or-content task: the margin only seems to play a constant role when the triplet evaluator is used. For the triplet evaluator the use of the limiting dataset brings down the performance slightly. Lastly, using a contrastive loss makes models score worse than using a triplet loss.

4.2.5 *Simplicity*

For the last task, there is only a trend in the margin when using triplet loss. The other factors do not have a consistent pattern. For the STEL-or-content task, the margin also only plays a role for the triplet loss. Additionally, the use of a contrastive loss decreases the performance.

4.2.6 *General trends in the STEL task*

Overall, the trend is that the triplet loss works better for every STEL-or-content task. The contrastive loss works better for the original task when only characteristics

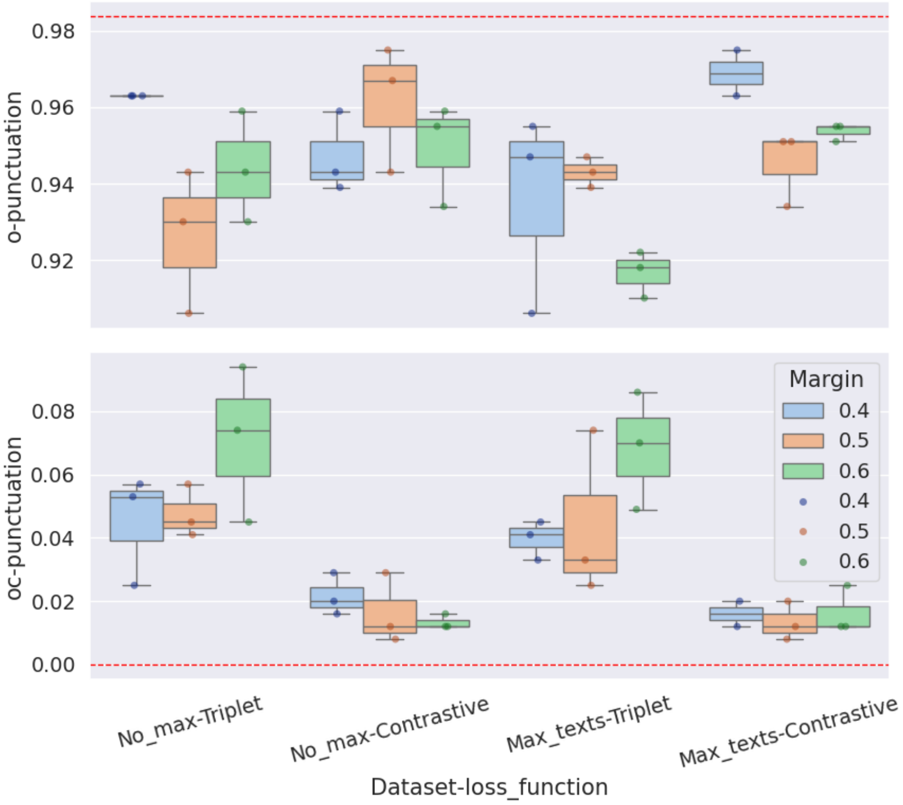


Figure 8: Boxplots and stripplots of the scores the fine-tuned models on the STEL punctuation tasks. The X-axis represents the dataset used to train on and the used loss function. The red dotted line is the performance of the base model.

are changed (capitalization, contraction, punctuation). This difference disappears for the dimensions.

4.3 *Interactions between parameters*

To see if changing the parameters had any significant influence on the performance of the models, a three-way ANOVA was performed to compare the effects of the margin values, loss functions, and dataset on the test score of the CAV task and the STEL tasks. If there are significant interactions, it means that utilizing a certain parameter value is beneficial for the improvement of the model on the task at hand.

Before the test was carried out, the data was checked for homogeneity to not violate the assumptions of homogeneity (Navarro, 2015, p.517). This was done with Levene's test to assess the assumption of homogeneity of variances for the test score across the hyperparameters. This test indicates that variances are equal across groups, $F(11, 24) = 0.598, p = 0.841$ meeting the homogeneity assumption of a three-way ANOVA.

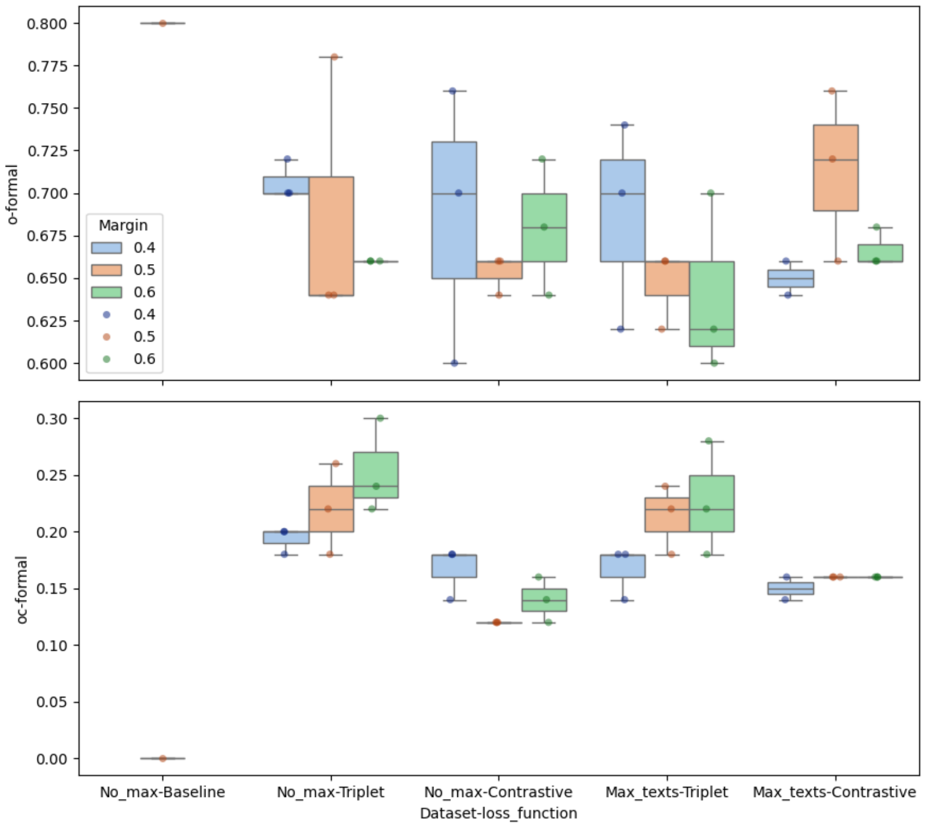


Figure 9: Boxplots stripplots of the scores the fine-tuned models on the STEL formal/informal tasks. The X-axis represents the dataset used to train on and the used loss function.

The normality of residuals was also checked (Navarro, 2015, p.518). This was done using the Shapiro-Wilk test for normal distribution. The normality was met for all groups except for the models fine-tuned using the no maximum limit on texts per author dataset with the contrastive loss and margins of 0.4 and 0.6. This means that not all data adheres to the normality assumption. The most probable reason for this is that there are only three results per group meaning that one outlier can already cause the data to be skewed. Considering that every other group adheres to this norm as well as taking into consideration that using a three-way ANOVA is the best fitting statistical test. I opted to still use this method while being cautious about drawing conclusions about the data.

The three-way ANOVA reveals that there is a significant effect of loss function on the test score $(F(1,23) = 6.49, p < .005)$. This indicates that a contrastive loss is statistically significantly more effective than a triplet loss. For the other factors, no significant effects were observed. The same is true for the interaction effects.

When looking at the quadruple STEL tasks, there are only two interactions that are found to be statistically significant through the use of a three-way ANOVA.

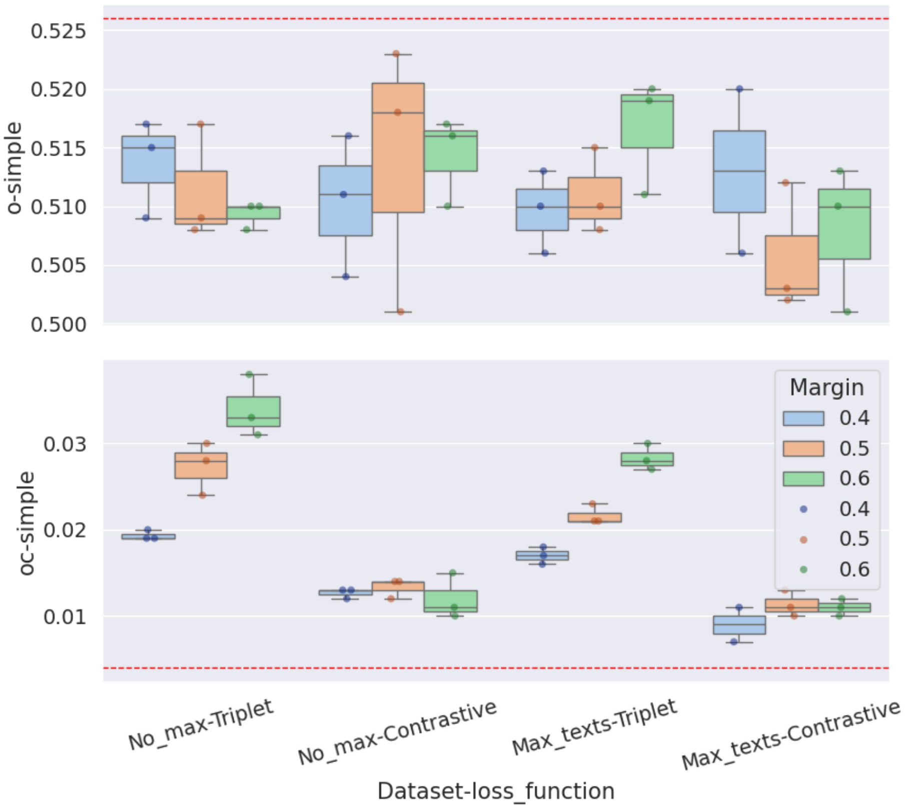


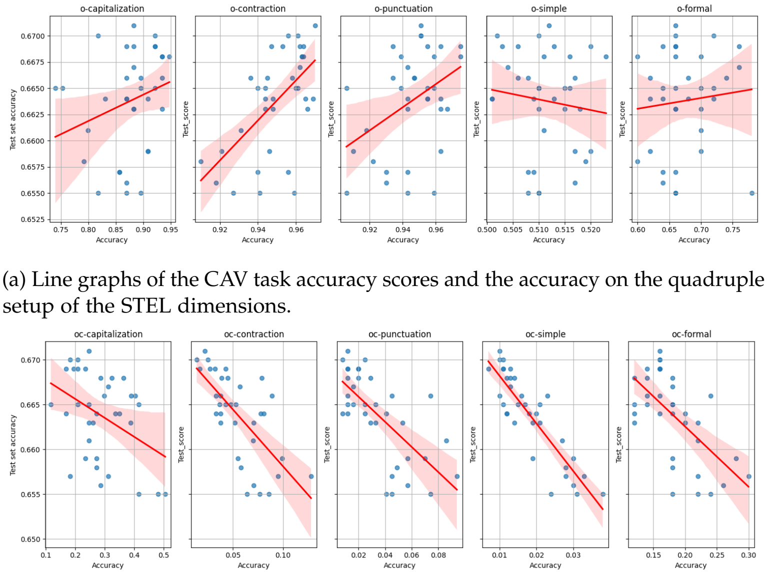
Figure 10: Boxplots and stripplots of the scores the fine-tuned models on the STEL simple/complex tasks. The X-axis represents the dataset used to train on and the used loss function. The red dotted line is the performance of the base model.

Both stem from the punctuation task. The first is the interaction between the loss function and the dataset. $(F(2,23) = 9.14, p < .005)$. There are no significant effects on the parameters separately, meaning that the loss function and dataset do not have a significant influence when changed alone. The second interaction is the dataset-margin-loss function interaction $(F(2,23) = 7.72, p < .005)$ on the punctuation task. This means that the loss function-margin interaction does not work in the same way for every margin. This can be seen in Figure 8 where the change of loss function or dataset does not lead to changes in the same direction for all margin values.

For the STEL-or-content task, the three-way ANOVA tests only found the simple task to be influenced by the change in factors. These factors are the loss function $(F(1,23) = 20.38, p < .001)$ and dataset $(F(1,23) = 4.28, p < .005)$ for the simplification part of the STEL test. There are also significant interaction effects between the loss function and the margin value $(F(2,23) = 8.75, p < .005)$ for the simplification task as well. This means that both the dataset and the loss function influence the model's performance independently while the margin function should only be considered in combination with the loss function. For all other tasks and all other factors and interactions, no significant values were found.

4.4 *The correlation between the CAV and STEL task*

To see if there is a correlation between the performance on the CAV test and the STEL test, the scores of the models on the test set are compared to the scores of the models on the STEL tasks. A comparison is made for each task separately. The results of these comparisons are shown in Figure 11.



(a) Line graphs of the CAV task accuracy scores and the accuracy on the quadruple setup of the STEL dimensions.

(b) Line graphs of the CAV task accuracy scores and the accuracy on the triplet setup of the STEL dimensions.

Figure 11: Accuracy scores the fine-tuned models on the CAV test set compared to the scores achieved on the different datasets of the STEL tasks. **Counterintuitively the CAV scores are represented on the y-axis and the STEL scores on the x-axis.**

First, the data is checked on normality to determine if a Pearson test or Spearman correlation should be used. To test for normality a Shapiro-Wilk test is used on all of the STEL tasks as well as on the test score. The Shapiro-Wilk test indicated that the test set scores was significantly different from a normal distribution, $W = 0.93, p = 0.02$. This means that the Spearman correlation test should be used for all comparisons. The results of these correlations tests can be found in Table 8.

From these results, we can see that all the results on the STEL-or-content tasks are negatively correlated to the results on the CAV tasks. The strength ranges from the weakest related task, capitalization to the strongest, simplification. There is also a correlation for the original STEL contraction task. The other tasks are not correlated to the performance of the CAV task.

STEL-task	Correlation	p-value
o-capitalization	**0.353**	**0.037**
o-contraction	**0.545**	**7e-4**
o-punctuation	0.311	0.069
o-simple	-0.095	0.586
o-formal	0.126	0.471
oc-capitalization	**-0.363**	**0.032**
oc-contraction	**-0.699**	**3.0e-6**
oc-punctuation	**-0.692**	**4.2e-6**
oc-simple	**-0.852**	**8.7e-11**
oc-formal	**-0.604**	**1e-4**

Table 8: Spearman coefficients calculated on the results on the CAV task and the results on the STEL tasks where o represents the original STEL task and oc represents the STEL-or-content task.

5 DISCUSSION

In this section, the implications of the results will be discussed and the research questions will be answered. The results will also be contrasted to the findings of Wegmann and Nguyen (2021) and Wegmann et al. (2022) who presented similar methods as applied in this thesis. Furthermore, limitations that might have influenced the results of this study will be discussed as well as some suggestions for future work.

5.1 *Insights and interpretations of the results*

In this subsection insights and interpretations of the results will be given. The results demonstrate that the methods employed in this thesis have been effective in generating Dutch linguistic-style embeddings which falls in line with previous research into style embeddings for English.

First, the results of the contrastive authorship verification task (CAV) will be discussed and contrasted. The created models generally follow earlier scientific findings and show similar improvement when compared to their English counterparts. Secondly, the STEL task results are discussed. These results are harder to compare directly to previous research, but the results seem to be coming up short when compared to English variants.

5.1.1 *CAV task*

The best-scoring model increased the performance on the CAV task from the baseline (RobBERT) of 52.2% to 67.0%. This is an increase of 14.8 percentage points. This increase shows that it is possible for a Dutch BERT-like model to learn to perform the task of contrastive authorship verification. The low standard deviation of 0.001 further indicates that this configuration can learn the task consistently and that this score is not an outlier. This score is comparable to previous research.

When comparing these results to similar research for the CAV task in the English language (Wegmann et al., 2022), the improvement the two models make relative to their base models is comparable. Their base model, RoBERTa achieves a score of 63% on the CAV task while their best-scoring fine-tuned model reaches an accuracy of 77%. This means there is an increase of 14 percentage points compared to the 14.8 percentage points reported in my research.

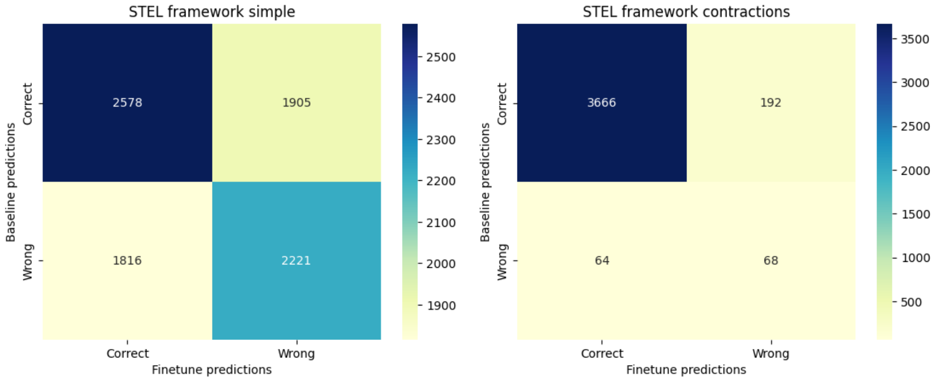
Only one of the parameters tried for training the model is statistically significant. Contrastive loss yields better results than triplet loss. The use of other parameters, although resulting in differing scores, did not result in a significantly different model performance. This is especially interesting for the use of a different dataset, where it would be expected that limiting the number of times an author can contribute a text to the dataset would introduce more stylistic variability to the model and thus improve its score. There can be large differences between models which only differ in the use of dataset. For example, the models that use a triplet loss and a margin of 0.5, see an increase of 0.9% when the dataset with maximum texts is used. Overall, this outlier is not indicative of the rest of the data, but it does show that there are settings where the use of a different dataset can yield significantly positive effects.

5.1.2 *STEL tasks*

In this part of the thesis, the results for both STEL tasks will be discussed and interpreted. This will be done for the original STEL task with a quadruplet set-up and for the STEL-or-content task which has a triplet setup. For this part, we use RobBERT-base as a baseline and our results are compared to the results of Wegmann et al. (2022). Not all STEL tasks are performed in both pieces of research. Therefore, only the formal, simple, and contraction tasks are compared and contrasted. The other tasks will be discussed but they cannot be compared to previous research.

When comparing this to the findings of Wegmann et al. (2022), I decided to use the scores of the finetuned RoBERTa model with no content control for their model since this is the most similar to my fine-tuned model. For my data, I used the model that achieved the best average score for all tasks. This was calculated by taking the individual scores on the tasks and taking this average to keep in mind the large difference in test instances ranging from $N = 50$ to $N = 3990$. The model that performed best uses the original dataset with no text limit and has a triplet loss function and a margin of 0.6. The results can be found in Table 9. Regarding the original STEL task, the results differ between the two studies. For contraction both fine-tuned models lose about 2 to 3 percent points in score, meaning that both models forget an equal amount of information. In the fine-tuned RoBERTa model, the score improves on the formal dataset while RobBERT forgets quite some information and sees a loss of 14 percent points. Lastly, the scores of the simple test go down for both fine-tuned models, but RoBERTa's score declines more than RobBERT's does.

Although the overall score for all STEL tasks (both original and STEL-or-content) is the best for the model with a triplet loss, a margin of 0.6, and a dataset that does not limit the number of texts per author, it does not mean that



(a) Heatmap of the simple STEL task

(b) Heatmap of the contraction STEL task

Figure 12: Heatmaps of the instances that the base model and the best scoring fine-tuned model got correct and wrong on the punctuation and simple/complex dimensions of the STEL task

it performs the best on all STEL tasks. When looking at Table 7 you can see that this model does not score the best in any of the original tasks when compared to the fine-tuned models. When averaging the scores in the same manner as above, but focusing solely on the original task, the best-performing model employs a contrastive loss with a margin of 0.4 and no limitations on the number of texts in the dataset. To illustrate how much a model can learn and forget, this model will be used for the following section.

When looking at Figure 12, we can see how many instances are learned and unlearned for the simple and contraction task. It can be seen that the model does indeed learn something compared to the base model, but the problem is that it forgets more than it learns. Leading to a decrease in accuracy. When put into percentages, it means that finetuning a model leads the model to learn 21.3% of the instances of the simple task but it forgets another 22.4%. For the contractions, it learns 1.6% and forgets 4.8%.

Model	Contraction		Formal		Simple	
	o	o-c	o	o-c	o	o-c
RoBERTa-base	1.0	.00	.83	.09	.73	.01
Fine-tuned RoBERTa	.98±.04	.00±.00	.85±.00	.50±.02	.56±.01	.04±.01
RobBERT	.97	.01	.80	.00	.53	.00
Fine-tuned RobBERT	.94±.01	.09±.01	.66±.00	.25±.03	.51±.00	.03±.00

Table 9: Accuracy score of base models and best performing fine-tuned models for the three STEL tasks that overlap between this thesis and Wegmann et al. (2022)'s paper.

Considering the triplet set up of the STEL task, the scores can also be found in Table 9. For the contraction task, the base models perform similarly but the finetuned RoBERTa model cannot learn anything whereas the RobBERT model can. The simple task is easier for the base RoBERTa model than for the RobBERT-base model. RoBERTa also improves more when fine-tuned than RobBERT does. For the simple task, the performances of the base and fine-tuned models are similar.

5.2 *Analysis per STEL task*

To discover what a model forgets when it is fine-tuned, it would be useful to add labels to the task data. Hereafter, I will discuss my recommendations for the STEL tasks to potentially find out what causes the models to unlearn. Thereafter, a possible solution for the unlearning is presented. By identifying what the model unlearns in this current research setup, it can later be compared to the novel approach that is not tried in this thesis.

5.2.1 *Capitalization*

The capitalization task is the only one where the base model can outperform the fine-tuned models on both the triplet and quadruple setups. Since capitalization is the only difference in these sentences, it suggests that the fine-tuned models may have unlearned the distinction between fully capitalized words and those in lowercase or standard casing.

Regarding further analysis, the sentences in the tasks could be checked on how many words were changed from fully uppercase to fully lowercase how many were changed to still start with a capital letter, and how many stay capitalized (because they are acronyms that should be capitalized). This might help classify the mistakes the fine-tuned models make and determine what the model unlearns.

5.2.2 *Contraction*

For the contraction dataset, data has already been provided indicating which contractions occur in which sentences. The results of this can be found in Table 10. What may be the most interesting from this data is that the model does not seem to solely learn one type of code pair e.g. there are two instances where the fine-tuned model learns to pair sentences in which the code *'t* (it) is changed in both sentences. At the same time, it also unlearns ten of these. This indicates that the model probably makes a decision based on something else rather than the contraction, which is interesting since nothing else in the sentences is changed except for these words.

5.2.3 *Punctuation*

The simplest example would be to add the number of exclamation marks that are replaced in an instance of the punctuation task. This is the only change that is made in this particular task which means that if there is a pattern to be found it must be connected to the number of exclamation marks. Especially since they are encoded based on the number of appearances each streak of exclamation marks makes in the training data. This means that, in theory, five exclamation marks might be encoded as one token while four exclamation marks might be encoded as the token for three exclamation marks followed by the token for one exclamation mark.

Wrong baseline & Correct finetune			Correct baseline & Wrong finetune		
Code 1	Code 2	# Occurrences	Code 1	Code 2	# Occurrences
’m	’t	5	m’n	’t	19
a’dam	’m	5	m’n	z’n	17
na’ja	z’n	5	’t	’t	10
a’dam	’t	4	’m	m’n	9
’t	z’n	4	z’n	z’n	9
hebb’ie	z’n	4	fluit’nt	’t	7
hij’s	m’n	3	’m	z’n	6
m’neer	’t	3	hij’s	z’n	4
a’dam	m’n	3	kwee’nie	’m	4
’t	’t	2	’k	m’n	4

Table 10: Comparison of the most common code pairs for instances where the base model and the finetuned model make different predictions for the STEL matching task on contractions. This means that code 1 is changed in the first sentence pair and code 2 is changed in the second pair.

It is known that points and commas receive special treatment in BERT’s encoding process due to their relatively frequent occurrence (Clark, Khandelwal, Levy, & Manning, 2019). Although no information on this was found about exclamation marks or RoBERTa (and therefore not about RobBERT), it seems likely that other punctuation marks also receive a different treatment than words in RobBERT’s encoding process.

If it turns out that there is no correlation between the number of exclamation marks that are replaced and the ability of the model to learn or unlearn, additional research would be required to find out if the words preceding the exclamation marks play a role in the model’s performance.

5.2.4 *Formal/informal*

The formal/informal sentences were rewritten without keeping track of any specific changes, this makes it difficult to determine what the model learns and unlearns. For future works, it might be interesting to keep track of the changes that are made in order to discover any patterns.

5.2.5 *Simple/complex*

The simple/complex dataset provides the changes made to the sentences for each task instance. These can be used to deduce if there are certain changes that are unlearned by the fine-tuned models in the same way as was done for the punctuation task. The added difficulty is that there might be several changes per sentence pair and thus no clear answer might be found about what the model unlearns and learns.

5.2.6 *Multitask learning*

The provided information, as illustrated in Figures 12a and 12b and supported by Table 10, demonstrates that the model both acquires and loses information relevant to creating stylistic embeddings. This is an example of catastrophic learning (Luo et al., 2023). To mitigate this, novel, unpublished research from the Blablablab[18] has demonstrated that fine-tuning the model with both a masked language modeling (MLM) task and a CAV task improves the model's performance. The base model is constructed with an MLM task, so including it in the fine-tuning phase as well as the CAV task means that the model's representations do not shift too much based on the new data that it is fed while still gathering new information from the CAV task. This method was attempted to be implemented, but due to timely constraints, it could not be included in this thesis.

5.3 *Relation between CAV and STEL task*

While it is true that the fine-tuned model performs worse on the quadruplet setup of the STEL task, it can be seen in Figure 11 that for all but one task, the performance on the task goes up as the performance on the CAV rises. For two tasks, this correlation is significant. The interesting thing is that the base models perform worse on the CAV while performing better on the STEL task. This is the exact opposite of what the fine-tuned models' data shows.

An explanation for this could be that there is not a linear correlation between the two but instead a higher-ordered one where the performance on the STEL task first drops if the model is trained in such a way that it only slightly improves on the CAV task after which it rises when the model is trained to function better on this task. This theory cannot be proven with the data gathered in this thesis and thus, further research is required.

Four out of five fine-tuned models outperform the base model when using the triplet setup. This is the opposite of the quadruple setup. What is also flipped is that performance decreases as the CAV score goes up. This correlation is significant for all tasks. It could be the case that the full graph of this correlation is also a parabola that starts low and rises as the CAV task improves somewhat after which it lowers again. However, this is not what we would expect. A conclusive answer cannot be given with the current data. Therefor, further research is required.

5.4 *Answering the research questions*

This thesis set out to answer the main research question: How can we create a representation of Dutch linguistic style using BERT-like language models and how can these representations be evaluated?

This was then subdivided into two subquestions which will be answered in this part.

[18] https://blablablab.si.umich.edu/

Research question 1: How can the methods of Wegmann et al. (2022) be adapted to fine-tune a model to learn Dutch style representations?

My research shows that the methods of Wegmann et al. (2022) are replicable in Dutch. By fine-tuning models on a BERT-like base model with Reddit texts as data, it is possible to achieve improvement of performance for these models when compared to the base. The most optimal combination of hyperparameters for the CAV task was using a contrastive loss with a margin of 0.4. Here, a dataset that maximized the number of texts an author was allowed to contribute to the dataset was limited. This resulted in an increase from an accuracy of 52.5% to one of 66.9%. This increase of 14 percent points is similar to the results from Wegmann et al. (2022) although it should be noted that similar, yet ultimately different techniques are applied in both studies.

Research question 2: How can the evaluation framework by Wegmann and Nguyen (2021) be adapted to evaluate Dutch style representations?

This thesis shows that replicating the dimensions used from the English STEL in Dutch is possible despite the stylistic differences between the two languages. Some dimensions can be copied in the sense that they change the sentences for the task based on the same characters, as in the contraction dimension. However, this does not imply that the two measure the same thing, as contractions serve different functions and occupy different positions in texts between the two languages. For others, it is the other way around, texts' different parts are changed to reach the same goal. For the simple/complex task, there was not a straight way to translate the English changes to Dutch but still, rules were found to achieve the same results. In conclusion, the translation of the STEL framework is possible, but research is required into each dimension to verify if it has the same function in Dutch as it has in English or research must be carried out to find aspects of a text that must be changed to achieve a stylistic shift.

5.5 *Limitations and future work*

In this subsection, the limitations of this study are discussed per part of the pipeline. At the same time, some potential future research areas are suggested.

5.5.1 *Dataset*

Regarding the data, it should be noted that the data was collected from Reddit, which means that it samples only a portion of the population's stylistic choices. Although many Subreddits were sampled, it is unlikely that the demographic of Reddit is representative of the whole Netherlands meaning that not all style choices are present in the representation. At the same time, finding representative data might be hard.

Another point about the dataset is that there are relatively many texts per author when compared to Wegmann et al. (2022) whose data only contained a maximum of nine texts per author. This indicates that the models were exposed to a greater number of authors, enabling them to better generalize.

5.5.2 *Data preprocessing*

One limitation / interesting subject for another study might be to research the style people use in different Subreddits. It could be the case that a user does not consistently use the style across all posts. This means that the training of the model might be limited when it is trained on texts that differ in style because they were posted on different Subreddits.

5.5.3 *Finetuning the model*

With regards to fine-tuning the model, it would be a possibility to also utilize different base models other than RobBERT. It could be discovered if RobBERTje's experiences a loss in performance and how big this loss would be. Additionally, it would also be possible to look at BERTje and see how this model performs on the task and if fine-tuning this model achieves a better result than the fine-tuned versions of RobBERT did.

5.5.4 *Creation of the STELNL evaluation dataset*

It could prove useful to change the Twitter data source. This data was quite useful since it contained a lot of informal language, but it was sampled from Twitter and only measures the model's ability to detect more extreme cases instead of more fine-grained style changes.

When creating the formal/informal dataset, it might be useful to create a ruleset to transform the sentence. When this is used, these rules can be used to detect what a fine-tuned model learns and unlearns compared to the base model.

A last recommendation would be to expand the dimensions of this thesis. The created dimensions are on the small side. Expanding the tasks would give a more reliable result. Some dimensions could also be expanded, the punctuation dimension could include question marks and ellipsis for example.

6 CONCLUSION

The first goal of this thesis has been to develop a method for optimizing the fine-tuning process of a BERT-like model using the contrastive authorship verification task (CAV). By exploring various datasets and parameter configurations, I have aimed to enhance the model's ability to embed linguistic stylistic features for the Dutch language. This has created the first model to explicitly try to represent these features for the Dutch language. The second goal has been to create a dataset that can be used to test the created embeddings on their ability to represent style.

The embedding training task for this thesis is derived from Wegmann et al. (2022) who use the same methodology for the English language. The results show that their methodology is transferable from Dutch to English. For the CAV task, both studies show that the increase in accuracy from baseline to fine-tuned model is 14 percent points, which means that the fine-tuned models learn equally much in both cases. The creation of the evaluation set was inspired by the STEL data frame from Wegmann and Nguyen (2021). Following their research, a Dutch equivalent of this data frame was constructed. Two of the tasks were created by other students of the Universiteit Utrecht under the supervision of Dr. Dong Nguyen and Anna Wegmann. The other three were created for this thesis. Of the five tasks, three were the same as in previous research. It should be noted that the results of these tasks cannot be directly compared to each other due to language differences. However, for the simple/complex and formal/informal tasks, both studies show that fine-tuning a model to represent style and testing this representation while controlling for content leads to an increase in performance for both models.

Though this thesis serves as a base for the research into Dutch style representations, much research is still to be conducted. For example, the question of why there seems to be a negative correlation between the fine-tuned models' scores on the CAV task and the triplet setup of the STEL tasks. Additionally, the Dutch STEL tasks could be expanded and enhanced both in terms of quantity and, in some cases, quality.

All in all, there are still many points on which the research into the representation of Dutch style can be improved. I hope that this thesis might serve as a solid base for those willing to contribute to the research into this topic.

REFERENCES

Abbasi, A., & Chen, H. (2005, September). Applying authorship analysis to extremist-group web forum messages. *IEEE Intelligent Systems*, *20*(5), 67–75. Retrieved from https://doi.org/10.1109/MIS.2005.81 doi: 10.1109/MIS.2005.81

Abbasi, A., & Chen, H. (2008, apr). Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Trans. Inf. Syst.*, *26*(2). Retrieved from https://doi.org/10.1145/1344411.1344413 doi: 10.1145/1344411.1344413

Adamovic, S., Miskovic, V., Milosavljevic, M., Sarac, M., & Veinovic, M. (2019). Automated language-independent authorship verification (for indo-european languages). *Journal of the Association for Information Science and Technology*, *70*(8), 858–871.

Ai, W., Lu, X., Liu, X., Wang, N., Huang, G., & Mei, Q. (2017). Untangling emoji popularity through semantic embeddings. In *Proceedings of the international aaai conference on web and social media* (Vol. 11, pp. 2–11).

Al-Rfou', R., Perozzi, B., & Skiena, S. (2013, August). Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of the seventeenth conference on computational natural language learning* (pp. 183–192). Sofia, Bulgaria: Association for Computational Linguistics. Retrieved from https://aclanthology.org/W13-3520

Altamimi, A., Clarke, N., Furnell, S., & Li, F. (2019). Multi-platform authorship verification. In *Proceedings of the third central european cybersecurity conference.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3360664.3360677 doi: 10.1145/3360664.3360677

Amos, O. (2008). The text trap. *The Northern Echo*, *27*.

Anbeek, T., & Verhagen, A. (2001). *Over stijl.*

Argamon, S., Koppel, M., Pennebaker, J. W., & Schler, J. (2009, February). Automatically profiling the author of an anonymous text. *Commun. ACM*, *52*(2), 119–123. Retrieved from https://doi.org/10.1145/1461928.1461959 doi: 10.1145/1461928.1461959

Barber, A. (2008). Idiolects. In *Stanford encyclopedia of philosophy.*

Biber, D., & Conrad, S. (2019). *Register, genre, and style.* Cambridge University Press.

Bloch, B. (1948). A set of postulates for phonemic analysis. *Language*, *24*(1), 3–46.

Boenninghoff, B., Nickel, R. M., Zeiler, S., & Kolossa, D. (2019). Similarity learning for authorship verification in social media. In *Icassp 2019 - 2019 ieee international conference on acoustics, speech and signal processing (icassp)* (p. 2457-2461). doi: 10.1109/ICASSP.2019.8683405

Brocardo, M. L., Traore, I., & Woungang, I. (2015). Authorship verification of e-mail and tweet messages applied for continuous authentication. *Journal of Computer and System Sciences*, *81*(8), 1429–1440.

Burrows, J. (2002, September). 'delta': a measure of stylistic difference and a guide to likely authorship. *Literary and linguistic computing*, *17*(3), 267–287.

Chomsky, N. (2002). *Syntactic structures.* Mouton de Gruyter.

Clark, K., Khandelwal, U., Levy, O., & Manning, C. D. (2019, August). What does BERT look at? an analysis of BERT's attention. In T. Linzen, G. Chrupała, Y. Belinkov, & D. Hupkes (Eds.), *Proceedings of the 2019 acl workshop blackboxnlp: Analyzing and interpreting neural networks for nlp* (pp. 276–286). Florence, Italy: Association for Computational Linguistics. Retrieved from https://aclanthology.org/W19-4828 doi: 10.18653/v1/W19-4828

Crystal, D. (2005). Speaking of writing and writing of speaking. Pearson Education.

Crystal, D., & Davy, D. (1969). *Investigating english style* (1st ed.). Routledge. doi: https://doi.org/10.4324/9781315538419

Daelemans, W. (2013). Explanation in computational stylometry. In *Computational linguistics and intelligent text processing* (pp. 451–462). Berlin, Heidelberg: Springer Berlin Heidelberg.

Danescu-Niculescu-Mizil, C., Gamon, M., & Dumais, S. (2011). Mark my words! linguistic style accommodation in social media. In *Proceedings of the 20th international conference on world wide web* (pp. 745–754).

de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., & Nissim, M. (2019, December 19). BERTje: A dutch BERT model. *ArXiv*.

Delobelle, P., Winters, T., & Berendt, B. (2020, November). RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the association for computational linguistics: Emnlp 2020* (pp. 3255–3265). Online: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2020.findings-emnlp.292 doi: 10.18653/v1/2020.findings-emnlp.292

Delobelle, P., Winters, T., & Berendt, B. (2022, April). RobBERTje: A distilled dutch BERT model. *arXiv preprint arXiv:2204.13511*.

Demir, N. M. (2016). Authorship authentication for twitter messages using support vector machine. *Southeast Europe Journal of Soft Computing*, *5*(2).

De Vel, O. (2000). Mining e-mail authorship. In *Proc. workshop on text mining, acm international conference on knowledge discovery and data mining (kdd'2000)*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, June). 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* (Vol. 1 (Long and Short Papers), pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from https://aclanthology.org/N19-1423 doi: 10.18653/v1/N19-1423

Diederich, J., Kindermann, J., Leopold, E., & Paass, G. (2003, may). Authorship attribution with support vector machines. *Applied Intelligence*, *19*(1–2), 109–123. Retrieved from https://doi.org/10.1023/A:1023824908771 doi: 10.1023/A:1023824908771

Dutta, S., Arunachalam, A., & Misailovic, S. (2022). To seed or not to seed? an empirical analysis of usage of seeds for testing in machine learning projects. In *2022 ieee conference on software testing, verification and validation (icst)* (p. 151-161). doi: 10.1109/ICST53961.2022.00026

Eder, M. (2011). Style-markers in authorship attribution: a cross-language study of the authorial fingerprint. *Studies in Polish Linguistics*, *6*(1).

Evans, V. (2017). *The emoji code: The linguistics behind smiley faces and scaredy cats*. Picador.

Fagel, S. (2009). The art of ordinary words and sentences: sentence complexity and madness in works by the dutch authors maarten biesheuvel and jan arends. In *Online proceedings of the annual conference of the poetics and linguistics association (pala).*

Fedulenkova, T. N. (2018). Stylistics. a resource book for students. *Language and Culture*(12), 118–123.

Feng, L. (2008). Text simplification: A survey. *The City University of New York, Tech. Rep.*

Firth, J. (1957). A synopsis of linguistic theory 1930-1955. *Studies in Linguistic Analysis, Special Volume/Blackwell.*

Forsyth, R. S., & Holmes, D. I. (1996, December). Feature-finding for text classification. *Literary and Linguistic Computing*, *11*(4), 163–174.

Fu, Z., Tan, X., Peng, N., Zhao, D., & Yan, R. (2018, Apr.). Style transfer in text: Exploration and evaluation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). Retrieved from `https://ojs.aaai.org/index.php/AAAI/article/view/11330` doi: 10.1609/aaai.v32i1.11330

Gao, T., Yao, X., & Chen, D. (2021, November). SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 6894–6910). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics. Retrieved from `https://aclanthology.org/2021.emnlp-main.552` doi: 10.18653/v1/2021.emnlp-main.552

Gero, K., Kedzie, C., Reeve, J., & Chilton, L. (2019, November). Low level linguistic controls for style transfer and content preservation. In *Proceedings of the 12th international conference on natural language generation* (pp. 208–218). Tokyo, Japan: Association for Computational Linguistics. Retrieved from `https://aclanthology.org/W19-8628` doi: 10.18653/v1/W19-8628

Giovanelli, M., & Mason, J. (2017). *The language of literature: an introduction to stylistics*. United Kingdom: Cambridge University Press.

Grant, T. (2013, June 5). Txt 4n6: method, consistency, and distinctiveness in the analysis of sms text messages. *Journal of Law and Policy*, *21*(2), 467–494.

Greer, D., Rice, M., & Deshler, D. (2014). Applying principles of text complexity to online learning environments. *Perspectives on Language and Literacy*, *40*(1), 9–14.

Halvani, O., Steinebach, M., & Zimmermann, R. (2013). Authorship verification via k-nearest neighbor estimation. In *Conference and labs of the evaluation forum.*

Harley, T. A. (2013). *The psychology of language: From data to theory*. Psychology press.

Hay, J., Doan, B.-L., Popineau, F., & Ait Elhara, O. (2020, November). Representation learning of writing style. In *Proceedings of the sixth workshop on noisy user-generated text (w-nut 2020)* (pp. 232–243). Online: Association for Computational Linguistics. Retrieved from `https://aclanthology.org/2020.wnut-1.30` doi: 10.18653/v1/2020.wnut-1.30

Hellinga, W. G., & van der Merwe Scholtz, H. (1955). *Kreatiewe analise van taalgebruik: prinsipes van stilistiek op linguistiese grondslag*. Noord-Hollandsche Uitgevers Mij.

Herrmann, J. B., van Dalen-Oskam, K., & Schöch, C. (2015). Revisiting style, a key concept in literary studies. *Journal of Literary Theory*, *9*(1), 25–52.

Retrieved 2024-11-01, from https://doi.org/10.1515/jlt-2015-0003 doi: doi:10.1515/jlt-2015-0003

Holmes, D. I. (1998). The evolution of stylometry in humanities scholarship. *Literary and linguistic computing*, *13*(3), 111–117.

Hoover, D. L. (2001, November). Statistical stylistics and authorship attribution: an empirical investigation. *Literary and linguistic computing*, *16*(4), 421–444.

Howard, B. S. (2009). Authorship attribution under the rules of evidence: empirical approaches in a layperson's legal system. *International Journal of Speech, Language & the Law*, *15*(2), 219–247.

Hu, Z., Lee, R. K.-W., Aggarwal, C. C., & Zhang, A. (2022, jun). Text style transfer: A review and experimental evaluation. *SIGKDD Explor. Newsl.*, *24*(1), 14–45. Retrieved from https://doi.org/10.1145/3544903.3544906 doi: 10.1145/3544903.3544906

Ierland, S. v. (2010). *Grammatical features influencing information structure. the case of l1 and l2 dutch and english*. Utrecht: LOT.

Iqbal, F., Khan, L. A., Fung, B. C. M., & Debbabi, M. (2010). e-mail authorship verification for forensic investigation. In *Proceedings of the 2010 acm symposium on applied computing* (p. 1591–1598). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1774088.1774428 doi: 10.1145/1774088.1774428

Jarvella, R. J. (1971). Syntactic processing of connected speech. *Journal of verbal learning and verbal behavior*, *10*(4), 409–416.

Joos, M. (1967). *The five clocks–a linguistic excursion into the five styles of english usage*. New York: Harcourt, Brace & World, Inc.

Juola, P. (2015, December). The rowling case: a proposed standard analytic protocol for authorship questions. *Digital Scholarship in the Humanities*, *30*(suppl_1), i100–i113.

Jurafsky, D., & Martin, J. H. (2024). *Speech and language processing*. Stanford University.

Khalid, O., & Srinivasan, P. (2020, May). Style matters! investigating linguistic style in online communities. *Proceedings of the International AAAI Conference on Web and Social Media*, *14*(1), 360-369. Retrieved from https://ojs.aaai.org/index.php/ICWSM/article/view/7306 doi: 10.1609/icwsm.v14i1.7306

Kingsley Zipf, G. (1932). *Selected studies of the principle of relative frequency in language*. Cambridge, MA & London: Harvard university press.

Kredens, K. (2003). Towards a corpus-based methodology of forensic authorship attribution: a comparative study of two idiolects. In *Palc'01: Practical applications in language corpora* (pp. 405–445). Peter Lang.

Labov, W. (1975). *Language in the inner city: Studies in the black english vernacular* (Vol. 53) (No. 4). Philadelphia, USA: University of Pennsylvania Press. Retrieved from https://doi.org/10.1177/089124167600400410 doi: 10.1177/089124167600400410

Labov, W. (1985). Hypercorrection by the lower middle class as a factor in linguistic change. In *Proceedings of the ucla sociolinguistics conference, 1964* (pp. 84–113). Berlin, Boston: De Gruyter Mouton. Retrieved 2024-08-12, from https://doi.org/10.1515/9783110856507-008 doi: doi:10.1515/9783110856507-008

Layton, R., Watters, P., & Dazeley, R. (2010). Authorship attribution for twitter in 140 characters or less. In *2010 second cybercrime and trustworthy computing* *workshop* (p. 1-8). doi: 10.1109/CTC.2010.17

Lee, C., & Barton, D. (2013). *Language online: Investigating digital texts and practices* (1st ed.). Routledge.

Leech, G. N., & Short, M. (2007). *Style in fiction: A linguistic introduction to english fictional prose* (No. 13). Pearson Education.

Li, J. S., Chen, L.-C., Monaco, J. V., Singh, P., & Tappert, C. C. (2017). A comparison of classifiers and features for authorship authentication of social networking messages. *Concurrency and Computation: Practice and Experience*, *29*(14).

Litvak, M. (2019). Deep dive into authorship verification of email messages with convolutional neural network. In *Information management and big data: 5th international conference, simbig 2018, lima, peru, september 3–5, 2018, proceedings 5* (pp. 129–136).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv*, *abs/1907.11692*. Retrieved from https://api.semanticscholar.org/CorpusID:198953378

Luo, Y., Yang, Z., Meng, F., Li, Y., Zhou, J., & Zhang, Y. (2023). An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Luyckx, K., & Daelemans, W. (2008, August). Authorship attribution and verification with many authors and limited data. In *Proceedings of the 22nd international conference on computational linguistics (coling 2008)* (pp. 513–520). Manchester, UK: Coling 2008 Organizing Committee. Retrieved from https://aclanthology.org/C08-1065

Marko, K. (2021). Exploring the distinctiveness of emoji use for digital authorship analysis. *Language and Law/Linguagem e Direito*, *7*(1-2), 36–55.

Marko, K. (2022). "depends on who i'm writing to"—the influence of addressees and personality traits on the use of emoji and emoticons, and related implications for forensic authorship analysis. *Frontiers in Communication*, *7*, 840646.

Matthews, R. A., & Merriam, T. V. (1993). Neural computation in stylometry i: An application to the works of shakespeare and fletcher. *Literary and Linguistic computing*, *8*(4), 203–209.

McMenamin, G. R. (2002). *Forensic linguistics: Advances in forensic stylistics* (1st ed.). CRC press.

Mendenhall, T. C. (1887). The characteristic curves of composition. *Science*(214s), 237–246.

Mendenhall, T. C. (1901). A mechanical solution of a literary problem. In *The popular science monthly.* LX.

Merriam, T. V., & Matthews, R. A. (1994). Neural computation in stylometry ii: An application to the works of shakespeare and marlowe. *Literary and Linguistic Computing*, *9*(1), 1–6.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In *International convention on learning representation.*

Miller, H., Kluver, D., Thebault-Spieker, J., Terveen, L., & Hecht, B. (2017). Understanding emoji ambiguity in context: The role of text in emoji-related miscommunication. In *Proceedings of the international aaai conference on web* *and social media* (Vol. 11, pp. 152–161).

Mosteller, F., & Wallace, D. L. (1984). *Applied bayesian and classical inference: the case of the federalist papers*. Springer Series in Statistics.

Navarro, D. (2015). *Learning statistics with r: A tutorial for psychology students and other beginners*. University of New South Wales.

Neal, T., Sundararajan, K., Fatima, A., Yan, Y., Xiang, Y., & Woodard, D. (2017, nov). Surveying stylometry techniques and applications. *ACM Comput. Surv.*, *50*(6). Retrieved from https://doi.org/10.1145/3132039 doi: 10.1145/3132039

Nicoladis, E., Duggal, A., & Besoi Setzer, A. (2023). Texting!!! attributions of gender and friendliness to texters who use exclamation marks. *Interaction Studies*, *24*(3), 422–436.

Olsson, J. (2007, December). Forensic linguistics: an introduction to language, crime and the law. In *language* (Vol. 83, pp. 899–901). Linguistic Society of America.

Oostdijk, N., Reynaert, M., Hoste, V., & Schuurman, I. (2013). The construction of a 500-million-word reference corpus of contemporary written dutch. *Essential speech and language technology for Dutch: Results by the STEVIN programme*, 219–247.

Pavlick, E., & Tetreault, J. (2016, 12). Erratum: "An Empirical Analysis of Formality in Online Communication". *Transactions of the Association for Computational Linguistics*, *4*, 565-565. Retrieved from https://doi.org/10.1162/tacl_a_00243 doi: 10.1162/tacl_a_00243

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

Rajaraman, A., & Ullman, J. D. (2011). *Mining of massive datasets*. Autoedicion.

Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 3982–3992). Hong Kong, China: Association for Computational Linguistics. Retrieved from https://aclanthology.org/D19-1410 doi: 10.18653/v1/D19-1410

Rivera-Soto, R. A., Miano, O. E., Ordonez, J., Chen, B. Y., Khan, A., Bishop, M., & Andrews, N. (2021, November). Learning universal authorship representations. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 913–919). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2021.emnlp-main.70 doi: 10.18653/v1/2021.emnlp-main.70

Sari, Y., Stevenson, M., & Vlachos, A. (2018, August). Topic or style? exploring the most useful features for authorship attribution. In *Proceedings of the 27th international conference on computational linguistics* (pp. 343–353). Santa Fe, New Mexico, USA: Association for Computational Linguistics. Retrieved from https://aclanthology.org/C18-1029

Savoy, J. (2020). Machine learning methods for stylometry. *Cham: Springer.*

Schroff, F., Kalenichenko, D., & Philbin, J. (2015, June). Facenet: A unified embedding for face recognition and clustering. In *2015 ieee conference on computer vision and pattern recognition (cvpr).* IEEE. Retrieved from http://

dx.doi.org/10.1109/CVPR.2015.7298682 doi: 10.1109/cvpr.2015.7298682

Smith, D., Spencer, S., & Grant, T. (2009). Authorship analysis for counter terrorism. *Unpublished Research Report, QinetiQ/Aston University*.

Stamatatos, E. (2007). Author identification using imbalanced and limited training texts. In *Proceedings of the 18th international conference on database and expert systems applications* (p. 237–241). USA: IEEE Computer Society. Retrieved from https://doi.org/10.1109/DEXA.2007.41 doi: 10.1109/DEXA.2007.41

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, *60*(3), 538–556.

Stamatatos, E. (2016). Authorship verification: A review of recent advances. *Res. Comput. Sci.*, *123*(1), 9–25.

Stutterheim, C. (1949). Modern stylistics. *Lingua*, *1*, 410–426.

Sundararajan, K., & Woodard, D. (2018, August). What represents "style" in authorship attribution? In E. M. Bender, L. Derczynski, & P. Isabelle (Eds.), *Proceedings of the 27th international conference on computational linguistics* (pp. 2814–2822). Santa Fe, New Mexico, USA: Association for Computational Linguistics. Retrieved from https://aclanthology.org/C18-1238

Taboada, M., Trnavac, R., & Goddard, C. (2017). On being negative. *Corpus Pragmatics*, *1*, 57–76.

Teh, P. L., Rayson, P., Pak, I., & Piao, S. (2015). Sentiment analysis tools should take account of the number of exclamation marks!!! In *Proceedings of the 17th international conference on information integration and web-based applications & services.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2837185.2837216 doi: 10.1145/2837185.2837216

Temnikova, I. (2012). Text complexity and text simplification.

Thurlow, C., Tomic, A., & Lengel, L. (2004). Computer-mediated communication: Social interaction and the internet. In *Online information review* (Vol. 2). SAGE Publications.

Toshevska, M., & Gievska, S. (2022). A review of text style transfer using deep learning. *IEEE Transactions on Artificial Intelligence*, *3*(5), 669-684. doi: 10.1109/TAI.2021.3115992

Tulkens, S., Emmery, C., & Daelemans, W. (2016). Evaluating unsupervised dutch word embeddings as a linguistic resource. *arXiv preprint arXiv:1607.00225*.

Turell, M. T. (2011). The use of textual, grammatical and sociolinguistic evidence in forensic text comparison. *International Journal of Speech, Language & the Law*, *17*(2), 211–250.

Tweedie, F. J., Singh, S., & Holmes, D. I. (1996). Neural network applications in stylometry: The federalist papers. *Computers and the Humanities*, *30*, 1–10.

Vandergriff, I. (2013). Emotive communication online: A contextual analysis of computer-mediated communication (cmc) cues. *Journal of Pragmatics*, *51*, 1-12. Retrieved from https://www.sciencedirect.com/science/article/pii/S037821661300057X doi: https://doi.org/10.1016/j.pragma.2013.02.008

Wegmann, A., & Nguyen, D. (2021, November). Does it capture STEL? a modular, similarity-based linguistic style evaluation framework. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp.

7109–7130). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2021.emnlp-main.569 doi: 10.18653/v1/2021.emnlp-main.569

Wegmann, A., Schraagen, M., & Nguyen, D. (2022, May). Same author or just same topic? towards content-independent style representations. In *Proceedings of the 7th workshop on representation learning for nlp* (pp. 249–268). Dublin, Ireland: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2022.repl4nlp-1.26 doi: 10.18653/v1/2022.repl4nlp-1.26

Williams, C. B. (1975). Mendenhall's studies of word-length distribution in the works of shakespeare and bacon. *Biometrika*, *62*(1), 207–212.

Wright, D. (2018). Idiolect. In *Oxford bibliographies in linguistics.* New York: Oxford University Press.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., & Callison-Burch, C. (2016, 07). Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, *4*, 401-415. Retrieved from https://doi.org/10.1162/tacl_a_00107 doi: 10.1162/tacl_a_00107

Yule, G. U. (1925). A mathematical theory of evolution, based on the conclusions of dr. j. c. willis, f.r.s. *Philosophical Transactions of the Royal Society of London. Series B, Containing Papers of a Biological Character*, *213*, 21–87. Retrieved 2024-08-12, from http://www.jstor.org/stable/92117

Yüzer, H. K. (2022). *Authorship attribution in turkish texts*. Artsürem.

Zhu, J., & Jurgens, D. (2021, November). Idiosyncratic but not arbitrary: Learning idiolects in online registers reveals distinctive yet consistent individual styles. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 279–297). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2021.emnlp-main.25 doi: 10.18653/v1/2021.emnlp-main.25

APPENDIX H: SUBREDDITS INCLUDED IN DATASET

- ADODenHaag
- AjaxAmsterdam
- Alkmaar
- Almere
- Amersfoort
- Amstelveen
- Amsterdam
- Arnhem
- BeermoneyNL
- Brabant
- Breda
- cirkeltrek
- D66
- Delft
- DenBosch
- DeStagair
- Deventer
- DNDNL
- Dordrecht
- Drenthe
- dutch
- DutchFIRE
- DutchHipHop
- efteling
- eindhoven
- enschede
- Eredivisie
- fcgroningen
- feyenoord
- Forum_Democratie
- FreeDutch
- Friesland
- Gelderland
- geldzaken
- Geschiedenis
- Gouda
- GroenLinks
- Groningen
- Haarlem
- HetHuisAnubis
- hilversum
- ik_ihe
- juridischadvies
- klokmemes
- knvb
- kopieerpasta
- learndutch
- Leeuwarden
- Leiden
- lekkerspelen
- limburg
- maastricht
- marktplaats
- medejongeren
- motorfietsen
- nederlands
- nietdespeld
- Nijmegen
- noordholland
- okemakkermaloot
- papgrappen
- partijvoordedieren
- PlaceNL
- Poldersocialisme
- Politiek
- PolitiekeMemes
- PSV
- PVV
- RMTK
- Roermond
- Rotterdam
- strips
- StudyInTheNetherlands
- TheHague
- thenetherlands
- Tilburg
- tokkiefeesboek
- tokkiemarktplaats
- TUDelft
- twente
- universityofamsterdam
- UniversityofTwente
- Utrecht

- utwente
- VeganNL
- Veluwe
- wageningen
- widm
- Zeeland
- zoetermeer
- zuidholland
- Zwolle