Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Pfaffenwaldring 5B
D-70569 Stuttgart

Bachelor Thesis

Effects of paraphrasing and demographic metadata on NLI classification performance

Miguel Marx Larre

Studiengang:	B.Sc. Informatik

Examiner:	Prof. Dr. Sebastian Padó
Supervisor:	Prof. Dr. Sebastian Padó

Registration Date:	25.11.2022
Submission Date:	25.05.2023

Abstract

Native language identification (NLI) refers to the task of automatically deducing the native language (L1) of a document's author, when the document is written in a second language (L2). Documents stem from different sources, but recently more documents are altered before publication through paraphrasing methods. This alteration changes the content, grammar, and style of the document, which inherently obfuscates the L1 of the author. In addition, the demographic metadata of the author, such as age and gender, may influence the performance with which an author's L1 may be detected. In this thesis, two corpora which provide necessary demographic metadata, the International Corpus of Learner English (ICLE) and the TRUSTPILOT corpus, are used to analyze the impact of paraphrasing and demographic factors in the context of NLI tasks. To analyze the effect of paraphrasing on a document, new versions of both corpora are created, which contain paraphrased versions of the documents contained. The effect is inspected using two state-of-the-art NLI systems to perform the task, while the results were analyzed using a regression analysis in combination with dominance analysis (DA). Paraphrasing was found to have a substantial influence in performance of NLI tasks, regardless of corpus, classifier, or paraphrasing method. The usual influence of demographic factors on NLI tasks could not be confirmed in this thesis. Regression analysis and DA allowed for a more profound analysis of the results, which allowed for findings regarding the influence of specific L1s on performance of NLI tasks.

Kurzfassung

Die Identifizierung der Muttersprache (NLI) bezieht sich auf die automatische Bestimmung der Erstsprache (L1) eines Autors anhand eines Dokuments, das in einer zweiten Sprache (L2) verfasst ist. Dokumente stammen aus verschiedenen Quellen, aber in letzter Zeit werden mehr Dokumente vor der Veröffentlichung durch Paraphrasierungstechniken verändert. Diese Änderungen beeinflussen den Inhalt, die Grammatik und den Stil des Dokuments und erschweren dadurch die Feststellung der L1 des Autors. Zusätzlich können demografische Metadaten des Autors wie Alter und Geschlecht die Genauigkeit der L1-Erkennung beeinflussen. In dieser Arbeit werden zwei Korpora, das International Corpus of Learner English (ICLE) und der Trustpilot-Korpus, verwendet, um den Einfluss von Umschreibungen und demografischen Faktoren im Zusammenhang mit NLI-Aufgaben zu analysieren. Um den Effekt von Umschreibungen auf ein Dokument zu untersuchen, werden neue Versionen beider Korpora erstellt, die paraphrasierte Varianten der enthaltenen Dokumente enthalten. Der Effekt wird mithilfe von zwei modernen NLI-Systemen analysiert, und die Ergebnisse werden durch Regression- und Dominanzanalyse (DA) untersucht. Es wurde festgestellt, dass Paraphrasierungen einen signifikanten Einfluss auf die Leistung von NLI-Aufgaben haben, unabhängig vom Korpus, Klassifizierer oder der verwendeten Paraphrasierungsmethode. Die übliche Auswirkung demografischer Faktoren auf NLI-Aufgaben konnte in dieser Arbeit nicht bestätigt werden. Die Regressionsanalyse und DA ermöglichten eine detaillierte Analyse der Ergebnisse und lieferten Erkenntnisse über den Einfluss bestimmter Muttersprachen auf die Leistung von NLI-Aufgaben.

Contents

1 Introduction

Native Language Identification (NLI) is the field within Natural Language Processing (NLP) which consists of the task of, given an English document, identifying the native language of the original author (L1), given that their native language was not English in the first place.

Usually, this is achieved via supervised machine learning, in which a classifier is trained on a corpus of English documents written by authors whose native language is not English.

NLI has been investigated in several different setups including many datasets, notably due to two NLI Shared Tasks for written documents in 2013 and 2017 (Tetreault et al., 2013) (Malmasi et al., 2017). In both, participants presented winning results using innovative technologies, such as stacked classifiers used by Cimino and Dell'Orletta (2017).

However, all existing work as of now focuses on original and unaltered documents. This does not represent the current state of published written documents, as tools such as Grammarly and LanguageTool are increasingly gaining new users. OpenAI has arguably caused the largest increase in documents which do not reflect the authors' original abilities. Their publicly available service ChatGPT has reached a million users within 5 days of its launch (Altman, 2022), generating new controversy about the legitimacy of texts that do not represent the user's abilities (Thorp, 2023). ChatGPT uses a large language model (LLM), which are capable of not only generating texts, but are generally able to perform a broad range of NLP tasks, such as text summarization, sentiment analysis, translation, or NLI. Services such as the three mentioned are used, among other things, to paraphrase a document. Often the goal is to alter the document in a manner that rectifies errors in the original document, or improves legibility and style. This way, the author of a document is able to produce a document of a quality that supersedes their true skill or hides mistakes made due to the native language of the author.

This circumstance creates a new difficulty for NLI classification tasks because the trained classifiers try to classify documents, which have already been altered. Be-

cause of this, the hypothesis of this work is that the mentioned AI systems influence the final document so much, that NLI becomes significantly more difficult.

In this thesis, we investigate how the performance of classifiers changes on NLI tasks when the documents to be classified are paraphrased by AI models, as well as the influence of demographic factors on unaltered versions of the documents. The effects of paraphrasing documents have not been researched in the context of NLI, which makes this the main novelty for this work.

To further understand how paraphrasing changes a classifiers' prediction, and to combat potential biases, the demographic properties age and gender of the author will be researched as well. After classifying the documents, the influence of the demographic properties will be investigated by determining how well a classifier performed on documents of a specific gender or age group. This way, the influence of the paraphrasing can be broken down to determine, how younger, older, male, or female authors are influenced differently, and if there is any bias present. We assume there to be a difference between all demographic groups, since topics of documents vary depending on the author's gender and age, as has been shown before by Hovy et al. (2015).

Given the motivation mentioned, this work targets two separate research questions. The first, "Do non-paraphrased and paraphrased corpora differ in difficulty in NLI, and how is this difference mediated by demographic factors?", aims at investigating how paraphrasing a corpus influences the performance of an NLI classifier. Since a potential change in performance on a paraphrased corpus will likely also be reflected by the demographic factors, we investigate how they behave in the different corpora. The second question, "Do classifiers transfer to other corpora which have been modified by paraphrasing, and how are demographic factors influenced by the transfer?", behaves similarly, but focuses on the transfer of a classifier across two corpora, of which at least one has been paraphrased.

Each experiment investigating a research question will make use of two corpora, and newly created corpora, which will be generated through paraphrasing the two original corpora using a language model by OpenAI.

To better analyze the results, and to extract more information on the behavior of demographic factors, we will use regression analysis and dominance analysis as has been done before in works such as Dayanik et al. (2022). This method of analyzing the influence of demographic factors on NLI has been researched very little, especially when combined with the paraphrasing aspect.

2 Technical Background

This section is intended to provide the necessary background knowledge to understand this work. To achieve that, a brief introduction to machine learning and its related technologies is given, along with a description of NLP and its subfield NLI.

2.1 Machine Learning

Machine Learning is defined as "a field of computer science that studies algorithms and techniques for automating solutions to complex problems that are hard to program using conventional programming methods" by Rebala et al. (2019).

Machine learning algorithms process datasets to predict the solution to a problem. The learning process is achieved through mathematical procedures, which vary depending on the implementation. Machine learning algorithms need data to learn from, which can be split into labeled and unlabeled datasets. Labeled datasets are datasets in which the solution to that problem or datapoint is already present. Unlabeled datasets, however, do not provide the solution to the posed problem.

Machine learning problems can be categorized into: supervised learning, unsupervised learning, and reinforcement learning.

In supervised learning, the algorithm learns from a labeled dataset. The algorithm can then learn patterns, and the importance of specific features within the dataset. Once the algorithm has learned from the data, it can make predictions on new data, where the answer is not provided already.

Unsupervised learning works similarly, but uses unlabeled datasets. The algorithm also learns underlying patterns from the data, without feedback about the correctness of a prediction. Clustering tasks are examples of unsupervised learning tasks, where data points are clustered based on only the features, without the need for labels.

Lastly, reinforcement learning is used in dynamic environments, such as video games or puzzles. The algorithm has to adapt to the environment changing, and

predict according to it. Reinforcement learning algorithms are provided with a value function, which serves as a heuristic for the predefined goal it should pursue. The algorithm's predictions are rated by the value function, and adapts according to its result.

The type of machine learning relevant to this work is supervised machine learning, as we only use labeled datasets.

2.2 NLP

Eisenstein (2019) defines Natural language processing (NLP) as "the set of methods for making human language accessible to computers". It mentions how NLP has become a part of our daily lives, since many applications make use of NLP. Some of the more present use cases are machine translation, text classification for spam detection, search engines, and dialog systems. However, the understanding and modelling of natural language as a whole without a specific application is also of interest, as it forms the basis for processing natural language using computers (Manning and Schutze, 1999). All of these applications require understanding of structure, wording, and context of the natural language, so it can be processed appropriately by a computer.

Due to the apparently unstructured nature of written and spoken language, texts are often transformed and represented in other ways, such as n-grams, or TF-IDF vectors. The datasets used to train NLP models are often constructed by collecting existing news articles, books, essays, or less formal texts such as social media posts.

2.3 NLI

Native language identification (NLI) is an NLP task that tries to determine the native language (L1) of a text's author. This is a challenging task, as it requires the algorithm to identify subtle differences in grammar, vocabulary, and style that are characteristic of the L1. Stehwien and Padó (2016) identified the use of "specially" as a mistake predominantly made by Spanish authors, at the same time, French

authors tend to misspell “example” as “exemple”. German authors stand out due to their introduction of subordinate clauses using a comma, as in “, that”.

NLI is interesting from a linguistic perspective, but also offers practical use. It can be used in forensics to determine the L1 of a suspect writing in English (Perkins, 2014). It can also be used to study language transfer, which happens when linguistic features from one language are applied to another, this often happens when learning a new language (Malmasi and Dras, 2014). More profound understanding of language transfer can then be used in a pedagogic context by language learning systems, to help remedy the effects of language transfer when acquiring a second language.

In the last decade, many new techniques to improve classifier performance have been established. The results of the first NLI shared task (Tetreault et al., 2013) show, that SVMs have always shown high accuracy on NLI classification tasks, mostly using word n-grams and POS n-grams as features. In 2017, the second NLI shared task (Malmasi et al., 2017) took place, again demonstrating the dominance of SVMs and n-grams.

In their work, Cimino and Dell’Orletta (2017), who were the winning participants of the 2017 NLI shared task (Malmasi et al., 2017), not only additionally used syntactic features, which were used in the 2013 already, just not by as many teams, but they also achieved the winning F1-score using stacked classifiers. Although many other entries performed similarly well, the ItaliaNLP Lab team outperformed other approaches, including traditional SVM architectures and ensembles.

Malmasi and Dras (2018) demonstrated how Tetreault et al. (2012) introduced the idea of classifier ensembles, and how it has been used since. This technique together with classifier stacking noticeably improves accuracy for NLI classification tasks, as can be seen in the methods used by the teams in (Malmasi et al., 2017).

2.4 TF-IDF

Documents as such cannot be processed very well by machine learning algorithms due to the apparently unstructured nature. However, methods exist to convert a

textual document into a matrix, which can then be processed by machine learning algorithms.

Term frequency-inverse document frequency (TF-IDF) is a weighting scheme used in information retrieval and NLP to convert purely textual data into numerical values in the form of a matrix. It assigns a weight to each term in a document, based on the importance of that term to the document and to the entire corpus.

The term frequency (TF) part of the weighting scheme measures the importance of a term to a particular document. Multiple variations exist, including forms in which TF is normalized. Schütze et al. (2008) calculates it using the logarithm as such:

$$tf(t,d) = \begin{cases} 1 + log(f_{t,d}) & \text{, if } f_{t,d} > 0 \\ 0 & \text{, otherwise} \end{cases}$$

$$f_{t,d} : \text{Count of term } t \text{ in document } d$$

This means that terms that appear more frequently in a document will be given a higher weight.

The inverse document frequency (IDF) part of the weighting scheme measures the importance of a term to the corpus as a whole. It is calculated as the logarithm of the total number of documents in the corpus, divided by the number of documents that contain the term. This means that terms that are rare across the corpus will be given a higher weight. There are different variants to calculate the IDF, one example includes avoiding a potential division by zero by adding 1 to the denominator.

$$idf(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

$$D : \text{Collection of documents}$$

$$N : \text{Total number of documents in collection} = |D|$$

$$|\{d \in D : t \in d\}| : \text{Count of documents in which term } t \text{ appears}$$

The TF-IDF weight of a term in a document is calculated by multiplying TF and IDF.

$$tfidf(t, d, D) = tf(t, d) \cdot idf(t, D)$$

This means that terms that are both frequent in a particular document and rare across the corpus will be given the highest weight.

TF-IDF weighting as a feature for NLI has been introduced in Gebre et al. (2013), and also showed improvements when used. It has been used consistently since its introduction.

2.5 Classification and SVMs

Machine learning tasks can be divided further, with one of its subfields being classification tasks. Rebala et al. (2019) defines classification tasks as "the problem of identifying the category to which an input belongs to among a possible set of categories".

This is most often used when data needs to be sorted into predefined categories. For example, a classification algorithm might be used to recognize different types of animals in a photograph. In that case, the predefined categories are the different animals.

Many techniques exist to perform this type of task, with neural networks, logistic regression, and support vector machines (SVMs) being the relevant ones for this work.

SVMs are a type of supervised learning algorithm that can be used for classification and regression tasks. The goal of an SVM is to find the optimal hyperplane that maximally separates the data points in a dataset, based on their class labels. These data points can be understood as vectors in a multidimensional space. There are many ways in which documents can be converted and represented as vectors in that space, with the recently introduced TF-IDF matrix being used often.

Given a dataset consisting of vectors $\vec{x}_i$, we want to find a separating hyperplane in the form of $\vec{w}^{\mathrm{T}}\vec{x} - b = 0$, where $\vec{w}$ is the normal vector of the hyperplane and b

is a bias term, which will be our decision boundary. To classify a datapoint $\vec{x}_i$, we compute $f(\vec{x}) = \text{sign}(\vec{w}^{\text{T}}\vec{x} + b)$, as this yields 1 or -1 depending on which side of the decision boundary the datapoint lies. Since we want to maximize the margins of the datapoints to the decision boundary, we want to maximize $\frac{2}{||w||}$, as this denotes the length of the two support vectors. To instead convert this into a minimization problem, we minimize $\vec{w} \cdot \vec{w}$ as is explained by Cortes and Vapnik (1995).

The result of this process is visualized in Figure 1, where the decision boundary separates all datapoints while having maximized the length of both support vectors.

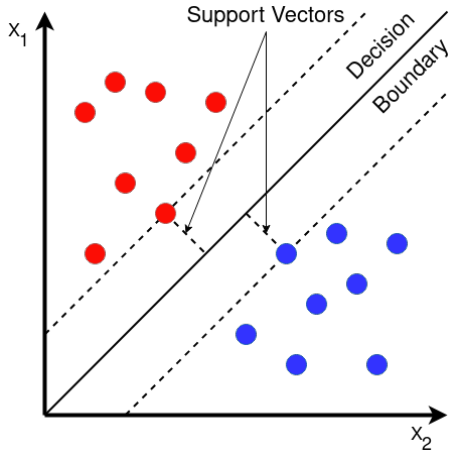

Figure 1: Visualization of SVM procedure. The decision boundary separates the data points by class, and maintains the maximum distance to the nearest point of each class.

An SVM is a binary classifier, which means that it can only distinguish between two classes. In the case of a multi-class classification problem, multiple SVMs can be trained, one for each class pair, essentially allowing SVMs to work as multi-class classifiers.

In contrast to classification by neural networks, SVMs stand out due to their low resource consumption and the lack of long and expensive training times.

2.6 Regression

Another prominent subfield of machine learning alongside classification is regression. Regression is defined as "an act of learning from existing data (can be past data) in trying to find relationships in the data" by Rebala et al. (2019).

Linear regression predicts a continuous numerical value based on the existing data using a model such as this one:

$$y \sim \alpha_1 x_1 + \cdots + \alpha_n x_n$$

Here the x_n are the values of the selected features, and α_n are the coefficients assigned to each feature. The coefficients α_n are commonly learned using methods such as least-squares estimation, which minimize the sum of the squares of the residuals. This way, the value of y can be predicted.

Logistic regression works similarly, but predicts the probability of the binary target variable y being 1 as opposed to 0. To achieve this, a similar model is used, but additionally the sigmoid function is applied to it. The coefficients are learned using maximum likelihood estimation (MLE), which may also be used to learn the coefficients of linear regression models.

$$P(y = 1) \sim \sigma(\alpha_1 x_1 + \cdots + \alpha_n x_n)$$
$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

More details about the characteristics of each model are provided in works such as Harrell et al. (2001). Similar to classification tasks, regression tasks can be applied to many fields, such as economics (Hellwig, 2014), which lie outside of computer science scopes.

2.7 Neural Network models

Neural network models are structures meant to resemble the human brain, they consist of many artificial neurons which are interconnected. The artificial neurons transform and process data from the input or previous neurons and output it to the next neuron or output. Neural networks typically consist of an input layer, one or multiple hidden layers, and an output layer.

2.7.1 Representation Learning

Neural networks rely heavily on the data they are provided to learn on. More precisely, they are also very reliant on the representation of the data they are provided, and on the features provided to them. The process of defining the features to be used often requires domain knowledge specific to the task, and is called feature engineering (Bengio et al. (2013), Liu et al. (2023)). Depending on the domain, the resulting features may vary strongly, as features for two different fields may not be interchangeable.

2.7.2 Word Embeddings

The most important feature for NLP tasks are word embeddings. Word embeddings are a representation of words in text as a vector in a multi-dimensional space. The main advantage of word embeddings is that they can capture semantic and syntactic information, such as context, similarity, synonymy, and antonymy. Word embeddings are obtained through machine learning, which generates the embedding for each word.

Crucially, word embeddings can be learned from unlabeled data. Devlin et al. (2019) describe how masked language modeling and next sentence prediction can be used to learn word embeddings without labeled data. During masked language modeling, a random word in a sentence is "masked" (by replacing it with a special token such as `[MASK]`), and the language model then predicts the missing word. Next sentence prediction is a binary task, in which the language model predicts, whether

a given sentence is likely to be followed by another. This training procedure helps the model learn the relationship between sentences and how they combine to form a coherent paragraph or document.

Word2vec (Mikolov et al., 2013) and the word embeddings produced by Google's BERT (Devlin et al., 2019) are both popular methods capable of producing word embeddings. However, the main difference is that BERT generates multiple word embeddings for the same word if the context differs to capture the context better depending on the sentence. This is useful for sentences containing polysemantic words, such as "Tennis is played using a green ball." and "I went to a ball dance yesterday.", where the word "ball" should be represented using different embeddings according to context.

Due to the nature of the vector-space representation of word embeddings, which already capture many features relevant for language processing, they are a promising method to improve performance on many NLP tasks.

2.7.3 Language Models

Language models are computational models trained on large amounts of text data, that can generate or predict text based on that training. They do this by assigning a probability to a sequence of words. This mechanic enables language models to perform tasks such as text generation and text classification.

Language models with many parameters, such as BERT with over 100 million parameters or GPT-3 with 175 billion parameters, are often called large language models (LLMs).

Liu et al. (2023) describes the two modern paradigms to create and train NLP language models. The first paradigm, "pre-train and fine-tune", is realized by pre-training a model with a fixed architecture on large amounts of data, while predicting the probability of the data observed. The resulting pre-trained language model can be adjusted to be applied on different downstream tasks by introducing additional parameters and fine-tuning them according to the requirements of the task. Because

of this, the language model does not need to be pre-trained for every new task, which reduces the training time significantly.

In the second paradigm, "pre-train, prompt, and predict", the model is not fine-tuned for a specific task, instead the tasks are changed to resemble original training data by formulating them as prompts. These prompts, which often consist of unfinished sentences, are then filled or continued by the language model.

2.7.4 LLM Architectures

Since the terms language model and neural networks cover many technologies and advancements, the most relevant architectures to the models used in this work will be introduced here.

Recurrent neural networks (RNNs) are "models of neural networks that are best suited for processing sequential data" (Rebala et al., 2019). Considering that many types of data, such as DNA or sentences, are sequential, RNNs are well-suited models for tasks related to those types of data. The main advantage of RNNs compared to other neural networks, is that they consider past data and their outputs, such as the output of a previous word in a sentence.

Self-attention is a mechanism that allows neural networks to focus on the most relevant parts of the input data. This is achieved by weighing each element of the input based on its similarity to other elements in the input. The neural network can then attend to the most important parts of the input. Although RNNs can also attend to other elements in the data, the main benefit of self-attention mechanisms, is that it can be influenced by elements that are not near to the processed element efficiently. RNNs can also be influenced by elements further away from the current element, even from future elements by using back-propagation, but require more computational time to achieve a similar performance to self-attention mechanisms. This makes self-attention an important improvement for NLP, since longer inputs, such as documents with multiple sentences, can now be processed with greater performance.

Transformers are a type of neural network architecture that utilizes the mentioned self-attention mechanisms introduced by Vaswani et al. (2017). Their benefit lies in the effective processing of sequential data, such as textual documents or time-series data. Transformers consist of an encoder, or a decoder, or both, depending on the implementation. The encoder uses self-attention mechanisms to convert the input, and the decoder generates the output given the previous representation.

2.8 BERT

BERT (Bidirectional Encoder Representations from Transformers) is an open-source (Google, 2023) state-of-the-art NLP model developed by Google (Devlin et al., 2019). It is designed to pre-train deep bidirectional representations from unlabeled text, where bidirectional means that it considers both left and right context.

The BERT model is a transformer-based model, which means it uses self-attention mechanisms to process input sequences and generate output sequences, as explained in 2.7. This makes the BERT model particularly well-suited for tasks that require a profound understanding of the underlying semantics, or context of the input text, such as natural language inference and question answering.

The BERT model is trained on large amounts of unlabeled text data, using masked language modeling and next sentence prediction, which were introduced in Section 2.7.2.

It is worth mentioning, that the standard BERT tokenizer only supports tokenizing 512 tokens, including the necessary `[CLS]` and `[SEP]` tokens. This is a limitation when tokenizing long texts, and approaches to remedy this have been proposed by Sun et al. (2019a), Mutasodirin and Prasojo (2021), and Beltagy et al. (2020).

The commonly used BERT-Base model (Turc et al., 2019) has been pre-trained on BookCorpus (Zhu et al., 2015) and English Wikipedia. Once the model has been pre-trained, it can be fine-tuned for a specific natural language task by adding a task-specific layer on top of the pre-trained model and training the whole model end-to-end on a labeled dataset. This can be seen in Figure 2, where many chained

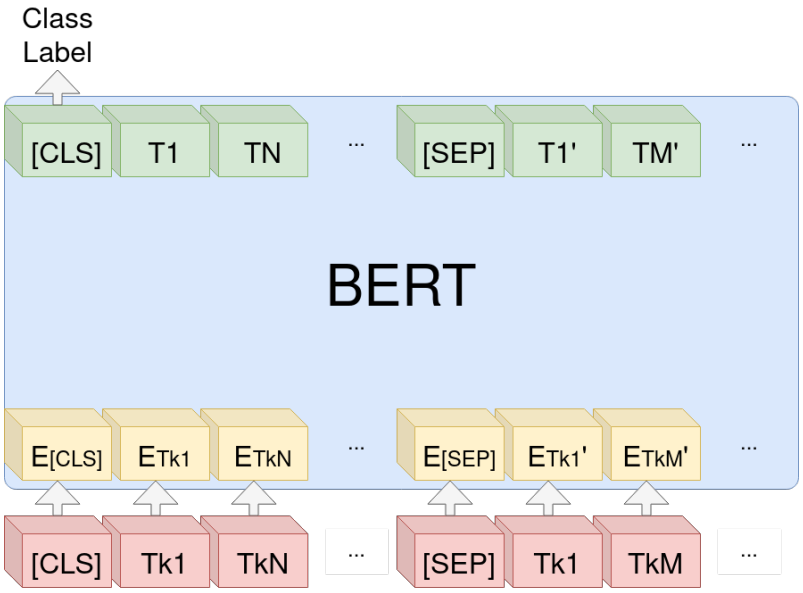

Figure 2: BERT fine-tuning schematic. With `E` as embeddings and `T` as contextual representation of an input token. Adapted from Devlin et al. (2019).

sentences are fed into the BERT model. This fine-tuning procedure allows the BERT model to adapt to the specific characteristics of the task, achieving state-of-the-art performance on a wide range of NLP tasks.

Since 2018, the BERT model by Devlin et al. (2019) has been used to match or surpass many accuracy milestones set in the past in the NLP field, although BERT requires more data to achieve better results than the traditional approaches Steinbakken and Gambäck (2020).

2.9 GPT-3

GPT-3 (Generative Pretrained Transformer 3) is a language model developed by OpenAI. It is a neural network that has been pre-trained on a large dataset of texts made up from a filtered version of CommonCrawl, WebText2, Books1, Books2 (two internet-based books corpora) and Wikipedia. GPT-3 can generate human-like text in a variety of different styles and formats.

GPT-3, like BERT, is based on the transformer architecture. Different from BERT, which is a bidirectional transformer model, it is an autoregressive language model, meaning that its output relies on its own previous outputs.

One of the key features of GPT-3 is its impressive size and scale. It has 175 billion parameters, making it one of the largest language models ever developed. For comparison, the previously introduced BERT-Base model, has 110 million parameters. The model reaches a size of 800 GB, which allows it to take into consideration 2048 tokens.

OpenAI (2023) lists GPT as being used by a wide range of applications on their website, including text generation, question answering, language tutoring applications, and machine translation. It has shown impressive ability to generate human-like text, and to perform a wide range of language tasks without any explicit programming or fine-tuning. The newest GPT model at the time of writing, GPT-4, was capable of passing the Uniform Bar Examination (UBE), a professional exam required to practice law in the USA (Katz et al., 2023).

However, it is not open-source and only available through a commercial API provided by OpenAI.

2.10 Evaluation

The performance of machine learning algorithms can be measured in various ways to determine whether the algorithm is sufficiently good at its task.

The performance of algorithms performing regression tasks can be measured using the R^2 measure, a measure of goodness of fit.

It indicates the proportion of variance in the dependent variable that can be explained by the machine learning model and is calculated as:

$$
\begin{aligned}
SS_{residual} &= \sum_i (y_i - f_i)^2 \\
SS_{total} &= \sum_i (y_i - \bar{y})^2 \\
R^2 &= 1 - \frac{SS_{residual}}{SS_{total}} \\
y_i &: i\text{-th value in dataset} \\
f_i &: i\text{-th value in predicted values} \\
\bar{y} &: \text{mean of dataset}
\end{aligned}
$$

When logistic regression models are used as a classifier, the R^2 measure cannot be applied, as the model does not predict a numerical value anymore. Instead, it is replaced by a pseudo-R^2 measure, which cannot be directly compared to the R^2 measure used in linear regression. Many pseudo-R^2 measures exist, as there is no agreed upon standard. A commonly used measure is the R^2_{McF} pseudo-R^2 measure by McFadden et al. (1973). It is defined as

$$R^2_{McF} = 1 - \frac{LL_{mod}}{LL_0}$$

where LL_{mod} is the log likelihood value for the fitted model and LL_0 is the log likelihood value for the null model.

To further measure classification algorithms, other metrics are used, the most common of which is accuracy, which is the fraction of datapoints which were correctly classified.

This, however, is a simple measure, which is why the F1 score is often a better indicator of good performance.

It is based on precision (P) and recall (R), which are defined as:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

TP : True Positive

FP : False Positive

FN : False Negative

The F1 score is then calculated as:

$$F1 = \frac{2PR}{P + R}$$

A high precision means that the algorithm is good at avoiding predicting wrong instances of a category, while a high recall means that the algorithm is good at finding most instances of a category. The F1 score takes both of these factors into account, and provides a single metric that can be used to compare the performance of different models.

The F1 score can take on a value between 0 and 1, where a higher value generally indicates a better performance.

2.11 Bias and Fairness

Bias is the preference over one group, or idea, over another, which then leads to incorrect or unfair conclusions for the disadvantaged party. It is often associated and studied in human thinking, where phenomena such as "anchor bias", which describes the phenomenon of humans relying too heavily on the first piece of information

received when making decisions, occur (Tversky and Kahneman, 1974). Bias can be exerted on many groups, but bias against demographic groups, such as women, is a very common problem in NLP (Blodgett et al., 2020) (Sun et al., 2019b) (Hovy, 2015). A common example of bias is women being associated with professions such as "homemaker", while men are more often associated with professions which are traditionally dominated by males such as "computer programmer" as has been found in Lu et al. (2020).

Machine learning algorithms tend to be susceptible to bias, as they learn from existing data, which can already be biased in itself, generating a biased algorithm (Meyer et al., 2020). The International Corpus of Learner English (ICLE) corpus used in this work has been analyzed in Brooke and Hirst (2012) and Brooke and Hirst (2013), which detected a high topic bias. They demonstrated that topic bias is a significant problem when paired with NLI. In an experiment, they proved that splitting the corpus into each topic causes the accuracy to drop significantly, demonstrating that most classifiers make their decisions based on words only present when writing about specific topics. Filtering the documents to remove words also present in the prompt for the document resulted to be difficult, since too many other words also indicate the original topic. The resulting algorithms then apply the underlying bias to the new tasks, propagating it further.

Hovy (2015) demonstrated, that demographic metadata such as age and gender clearly improve classification performance when a classifier is allowed to use them as additional features. He demonstrated slight improvements when making classifiers aware of sentiment, topic, or gender. It is to be noted that he also showed a clearly differentiated distribution of document topics between male and female authors.

The presence of bias implies the presence of unfairness, in which the group against which the bias is directed, experiences a form of disadvantage. Modern-day examples include machine learning algorithms for credit approval, which may discriminate against certain demographics, complicating access to credits to entire groups (Fuster et al., 2022).

2.12 AI writing support

Artificial Intelligence (AI) writing support is a rapidly evolving field that combines NLP and machine learning to assist writers in various aspects of the writing process. Very often, AI writing support systems aim to enhance the quality of writing by improving grammar, spelling, style, and tone. Grammarly and LanguageTool, are two commonly used tools for grammar checking. Ventayen and Orlanda-Ventayen (2018) shows that graduate students report improved writing and improved confidence during writing when using Grammarly.

Modern language models like OpenAI's GPT-3 and ChatGPT are transforming the use-case of AI writing support systems, and converting to generating roles, instead of just assisting. Thorp (2023) shows how these new systems generate entire passages of text, questioning the authorship of available texts.

But these language models create new ethical challenges, mostly stemming from the choice of training data. Flaws in the training data, such as gender bias, or racial bias, are likely to be present in outputs produced by the language model, propagating the biases further. This effect can also concern potentially sensitive data such as phone numbers or medical conditions in the training data. In some cases, misinformation may be spread by the language model when it was present in the training data, but in some cases the language model may even produce misinformation that is not present in the training data, which may even allege that innocent individuals committed serious crimes such as sexual misconduct (Verma and Oremus, 2023).

The effect most commonly discussed is the authorship of the documents because the authors of the documents in the training set are not quoted, however, the texts that the language model produces may contain entire passages of texts written by those authors. Since the authors are not credited and the producing texts may contain passages from a different author, the produced texts may be plagiarized without the user's knowledge.

Preventing some of the mentioned issues is easy, such as prohibiting the language model to output phone numbers, but most of the problems stem from data in the training set, that are only detected as problematic after human judgement.

These problems will continue to persist, due to the difficulty of detecting whether a document of the training set contains problematic data and due to the difficulty of detecting if produced content by the language model is misleading or plagiarized.

3 Data

Due to the nature of the experiments, which require corpora with demographic metadata along with a non-English native author, not many corpora fulfill all those requirements. This is to be expected, as a corpus containing that much information about its author raises privacy concerns. Table 1 shows some of the more commonly used corpora in the NLI field. Each entry is supplemented with the type of document present in the corpus, along with the information whether the required metadata, L1, age, and gender, are present, or can be deduced, from the corpus.

As can be seen from the table, only five of the selected corpora provide the necessary demographic data. Out of those five, the Artie Bias Corpus does only provide an accent of the author, which is not enough to determine the true L1 of the author. crowd-enVENT does provide some data about the author's origin, but since it is only as specific as "European", or "North American", no L1 can be deduced either. The Hate Speech Twitter corpus, does only provide the race of the author, which cannot be used to determine their native language.

The only two remaining corpora are the ICLE corpus, and the Trustpilot corpus, as they provide the author's native language along with the demographic metadata.

Both corpora are now introduced, as they are the basis of the experiments of this work. In addition, the following procedures, in which the data is prepared, are illustrated in Figure 7.

3.1 ICLEv2

The International Corpus of Learner English (ICLE) (Granger et al., 2009) in its second version is a collection of 6085 essays written by students learning the English language. The students were high intermediate to advanced learners of English (approximately equivalent to B2-C2 in the Common European Framework (CEF) (Council of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division, 2001)), and had 16 different native languages.

Name	Content	L1	Age	Gender
ICLE (Granger et al., 2009)	Essays	Yes	Yes	Yes
TRUSTPILOT (Hovy et al., 2015)	Trustpilot reviews	Yes	Yes	Yes
Hate speech Twitter (Waseem and Hovy, 2016)	Tweets	Only race	Yes	Yes
crowd-enVENT (Troiano et al., 2023)	Stories about emotional experiences	Only broad region	Yes	Yes
Artie Bias Corpus (Meyer et al., 2020)	Subset of Mozilla Common Voice Corpus	Only accent	Yes	Yes
ACL-NLI (Stehwien and Padó, 2015)	ACL articles	Yes	No	No
EFCamDat (Geertzen et al., 2013)	Englishtown Submissions	Yes	No	No
ICNALE (Ishikawa, 2011)	Essays	Country of origin	No	No
Lang-8 (Brooke and Hirst, 2013)	Texts for correction	Yes	No	No
L2-Reddit (Rabinovich et al., 2018)	Reddit posts	From Reddit flair	No	No
TOEFL-11 (Blanchard et al., 2013)	TOEFL-Tests	Yes	No	No

Table 1: Common available corpora for NLI

The length distribution of the documents is displayed in Figure 3, where most essays are between 500 and 1000 tokens in length. It has been noted multiple times, that a major flaw of the ICLE corpus is its topic bias (Brooke and Hirst, 2012) (Brooke and Hirst, 2013), which may inadvertently create a new feature by which a classifier may classify a document. This means that features present because of the author's L1 are drowned out by word choices present exclusively because of the document's proposed topic. Brooke and Hirst (2013) showed that simple attempts to soften or eliminate those words have to be very extensive, and are not feasible as it requires manual intervention in every document. This work will use ICLE in an unmodified state, as its topic bias cannot be eliminated. Future evaluations will consider, that topic bias is likely to improve performance.

The corpus provides documents by authors of 20 different L1 categories. The categories `Other` and `Unknown` are dropped because the L1 cannot be determined from that information.

Chinese languages are represented using three categories in the ICLE corpus, `Chinese`, `Chinese-Cantonese`, and `Chinese-Mandarin`. `Chinese` does not indicate what Chinese language is represented, since China has many official languages, which are spoken in the country. Since it also provides a small sample size, the category is dropped. Although `Chinese-Mandarin` does refer to a specific language, only eight documents are present in the corpus and are therefore dropped.

In Figure 4, the age distribution in the corpus after filtering out all entries without necessary information, as displayed in Figure 7, is shown. In our case, necessary information means that a valid gender is present and that the age of the author lies between 10 and 80.

Since ICLEv2 is a learner corpus and most students learn a language in school or during the early stages of their career, most authors of ICLE essays are comparatively young, between 18 and 30 years of age. The gender of the authors in the corpus was exclusively recorded as male, female, or unknown, omitting documentation of non-binary genders, which therefore cannot be accounted for. 76.82% of the authors are male and the remaining 23.18% are female. We split the corpus into an

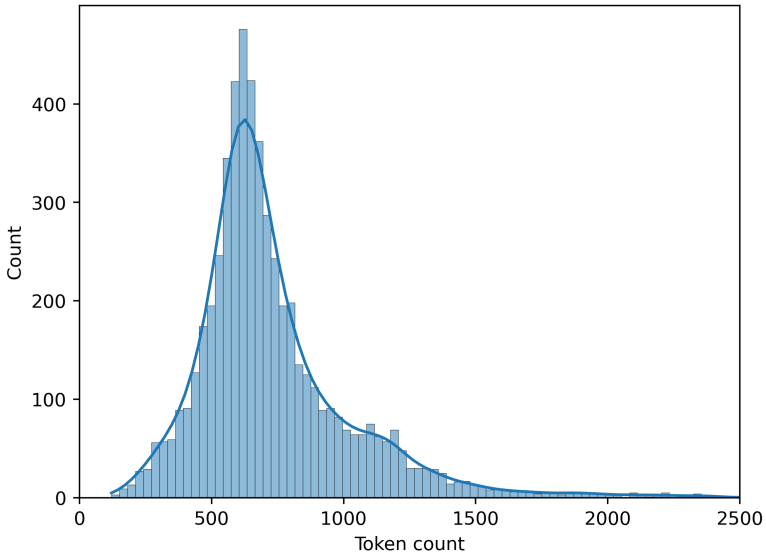

Figure 3: Token count distribution in the ICLE corpus.

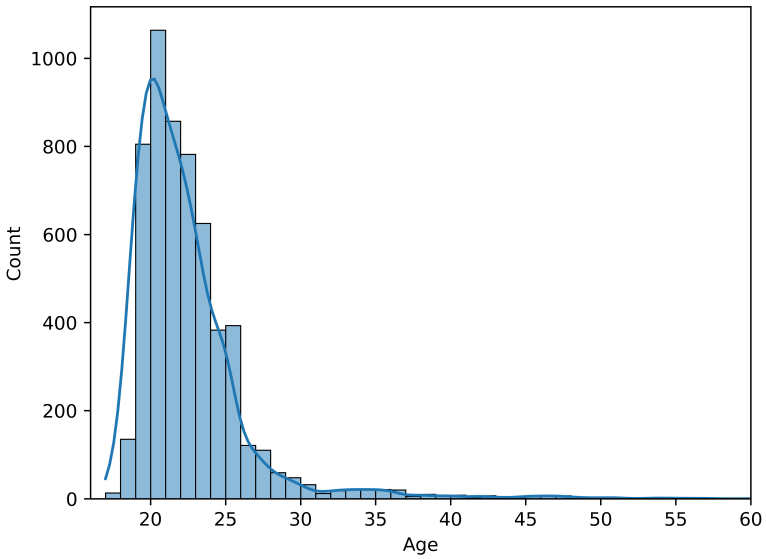

Figure 4: Age distribution in the ICLE corpus.

80% training set and a 20% testing set, as no predefined training and testing sets were given.

The essays cover many different topics, this is an excerpt of an essay written by a Spanish author answering the prompt "Are we on the right way?":

"My opinion may sound rather pesimistic, but I do think that as we have reached the wonderful "era of the computers", at the same time we have made ourselves cooler and more dependent on the very machines we have created. As far as technology is concerned , the problem is deep-rooted because nowadays since the childhood we are given very few chances of creating or imaging. There has been recently a boom in the selling of a type of games which I find specially harmful. [...]"

3.2 Trustpilot

The TRUSTPILOT corpus was created by Hovy et al. (2015) to demonstrate how existing reviews from large user-review sites, such as TRUSTPILOT, can be used to perform large-scale sociolinguistic studies such as this one. This corpus provides the review written by a user together with other required metadata. The metadata relevant to this work are the gender, age, and location data. The gender of an author was filled in when not already present, by inspecting the gender of all other authors of the same country, and determining whether a gender was predominant. Age was only supplied when already entered by the user. The location was determined by analyzing the location the user entered and applying a set of heuristics to deduce a latitude and longitude.

The corpus also contained documents by authors with English as their native language, and documents in the native language by German, Danish, and French authors. Since we want to use this corpus as a NLI corpus, we had to filter out all documents by the authors with English as a native language. Additionally, all documents in a language different from English were filtered using the `langdetect` library.

In this work, we also determine the L1 of an author by using the deduced location

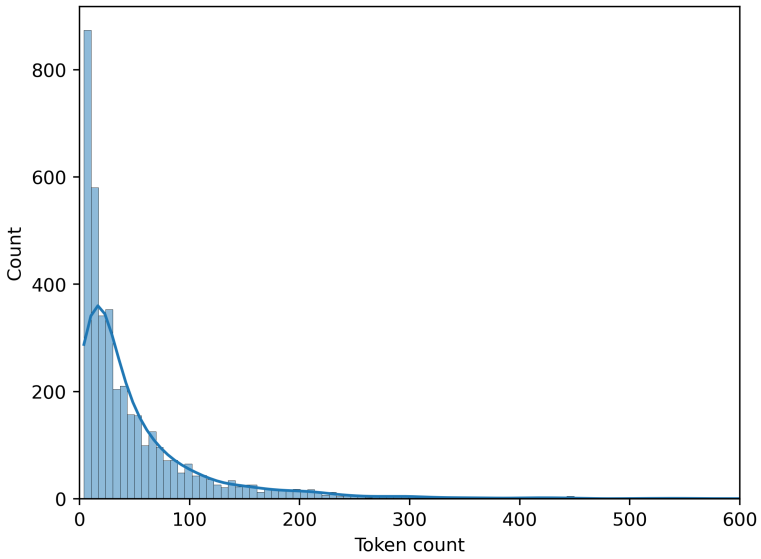

Figure 5: Token count distribution in the TRUSTPILOT corpus.

data, however as already noted by Hovy et al. (2015), this cannot be a guarantee for a correct L1. Errors will therefore exist, as reviewers who moved away from their birthing country or the place they grew up in will be assigned the wrong L1. We do not consider this an issue because we assume the share of authors to have moved away from their country to be small enough.

Since reviews of products or sites do not contain as much information as an essay, the token length distribution of the TRUSTPILOT corpus shown in Figure 5 shows that most reviews are shorter than 100 tokens, while only a small fraction is longer than 200 tokens.

In Figure 6 the age distribution in the corpus after filtering out all entries lacking the necessary information, as per Figure 7, is shown.

This corpus does not contain learners of English, but reviewers of many backgrounds. For this reason, the age distribution is not as concentrated, but spread out over all users of the platform. Similarly to the ICLE corpus, gender has been

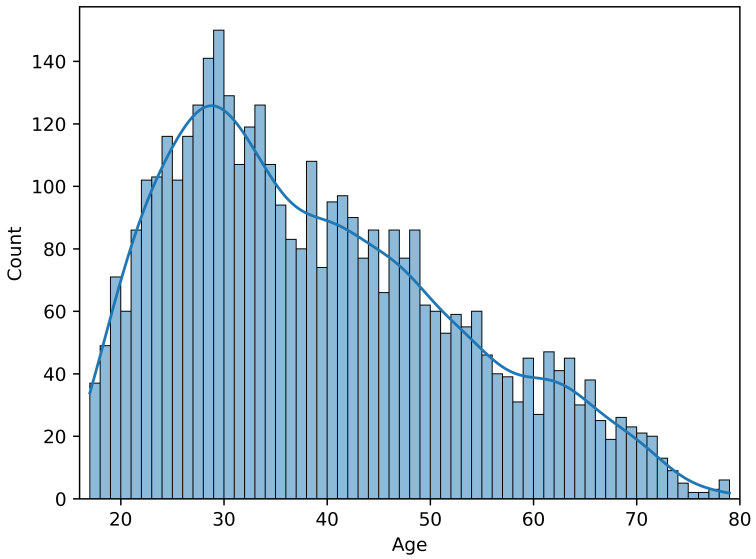

Figure 6: Age distribution in the TRUSTPILOT corpus.

recorded as a binary variable where most of the authors are male, with the exact distribution being 71.79% male and 28.21% female. This corpus has also been split into an 80% training set and a 20% testing set, for the same reasons mentioned in Section 3.1.

4 Methods

4.1 Workflow

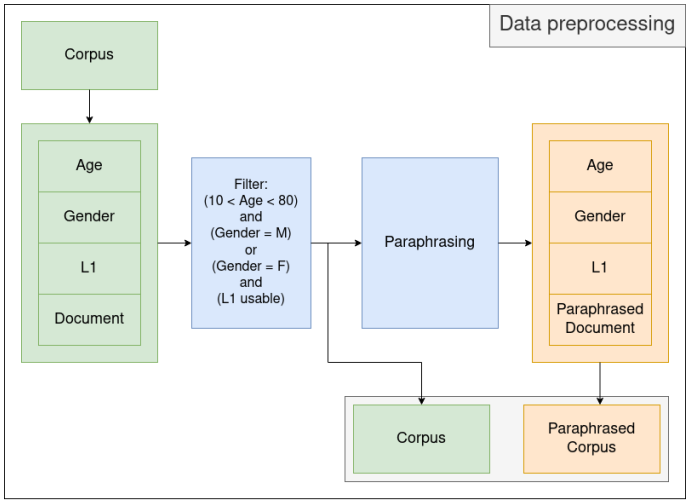

Figure 7: Process of data preparation including filtering and paraphrasing of a corpus.

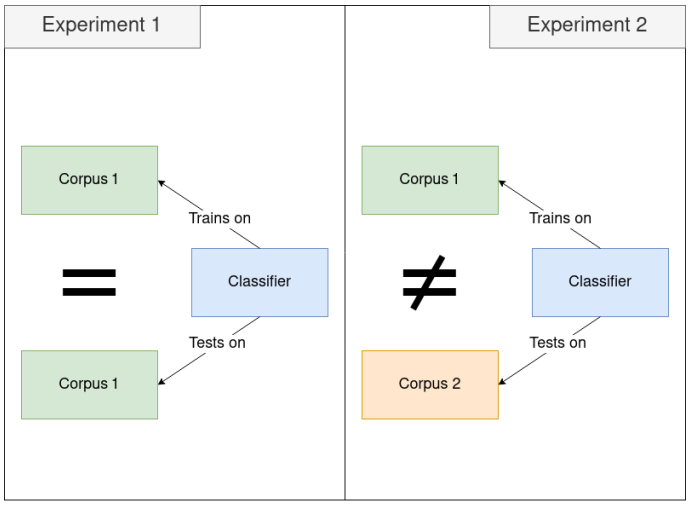

Figure 8: Showcase of the two experiments performed on corpora, where a classifier is trained on the same corpus, or different corpora.

To investigate the two research questions, we conduct two matching experiments. Before any experiment can be specified, we have to lay out our workflow, as we need to prepare the data, and specify our requirements before going into more detail about the experiments.

Figure 7 shows the structure of the dataset and any preprocessing necessary for the experiments. Section 3 already explains how both corpora were filtered to match our requirements for both experiments. In addition to the corpora presented, we need to create paraphrased versions of the corpora. The resulting set of corpora concludes the preprocessing of the data before the experiments themselves can be conducted.

After preparing the corpora to match our needs, we can further specify our experiments. Both experiments will be constructed similarly, as both experiments consist of a classifier classifying documents from one or more corpora. The main difference being that the second experiment will process different combinations of corpora, as is illustrated in Figure 8. The specific structure of each experiment is explained in the following two sections.

4.1.1 Corpus Creation

To create paraphrased versions of the documents provided by the corpora, we need to reorder and substitute words in a sentence, such that it sounds more proficient and correct. At the same time, orthographic and grammatical errors should not occur anymore, as they are replaced or corrected automatically. This is intended to reproduce the effects of existing tools used nowadays, such as autocorrection on phones, or paraphrasing tools used by students to create better sounding sentences in their thesis. It also constitutes our main hypothesis, that NLI tasks become harder when texts are paraphrased, as they already are in many produced documents today.

We will create three new versions of the corpora. As the documents in the Trustpilot corpus are comparatively short, as is shown in Table 5, only a single new corpus is created by paraphrasing each document sentence-wise. We created a paraphrased version of the ICLE corpus using the same method, but this raises

a problem with this method. When paraphrasing longer documents, such as those from the ICLE corpus, sentence-wise, the context of surrounding sentences is lost. Especially when treating with long essays from the ICLE corpus, the context of surrounding sentences is essential to the content of the current sentence. This is why we created a second paraphrased ICLE corpus using a more sophisticated method, which we will introduce in Section 5.1.

4.1.2 Experiment 1

The first experiment, which we will refer to as "Experiment 1", consists of a NLI classifier, a training set, and a testing set. This experiment aims at answering the question "Do non-paraphrased and paraphrased corpora differ in difficulty in NLI?". As mentioned before, we expect to encounter a significant increase in difficulty when using paraphrased corpora.

To conduct the experiment, the classifier will train on the training set, and will then be applied to the testing set. Both the training set, and the testing set stem from the same corpus. The available corpora are: ICLE, ICLE_P, ICLE_S, TRUSTPILOT, and TRUSTPILOT_P. This results in five testing-training combinations for each of the two used classifiers.

4.1.3 Experiment 2

Although this experiment, "Experiment 2", has the same structure as Experiment 1, it differs in the combinations in which the datasets are used. The classifier will now train on a dataset, but classify documents from a different dataset in addition to the documents from the training corpus. In the future, this will be denoted as `Corpus 1 - Corpus 2`, where `Corpus 1` is the corpus the classifier will train and test on, and `Corpus 2` is the corpus which will also be classified in addition to the first corpus. This creates training-testing combinations such as ICLE-ICLE_P, ICLE_P-ICLE, TRUSTPILOT-TRUSTPILOT_P, and TRUSTPILOT_P-TRUSTPILOT, for a total of eight testing-training combinations for each classifier.

By doing this, we aim at answering the question "Do classifiers transfer to other corpora which have been modified by paraphrasing?". We expect this experiment to yield more complex results, as the direction of classification will likely determine the performance.

4.2 Classification

Both experiments require an NLI classifier to work. To add a layer of comparability, we chose to use two different classifiers per experiment.

Since SVMs, introduced in Section 2.5, in combination with the TF-IDF weighting scheme have been the state-of-the-art classifiers for NLI tasks, we chose an SVM as our first classifier. However, as already mentioned in Section 2.8, the BERT language model is a modern language model based on a neural network structure. As the fundamentals of SVMs and BERT differ strongly, we chose BERT as the second classifier. Due to the promising performance of BERT and other neural network language models in NLP tasks, one might consider this comparison a comparison between what has been the state-of-the-art approach to NLI tasks and a technology which may become the new state-of-the-art in such tasks (Devlin et al., 2019).

4.3 Evaluation

Both of the experiments generate a lot of data to analyze, which has to be analyzed in different aspects. This leads to a more sophisticated evaluation procedure, which is strongly based on Dayanik et al. (2022). We will also use the regression-based analysis with dominance analysis (DA).

This evaluation procedure uses a regression model to investigate potential biases in the classification results by another classifier. This approach has multiple advantages over other approaches to analyze biases, such as the simple analysis of performance differences. In addition to simply detecting biases, this approach, among other advantages, also offers quantified results for each predictor variable present in the regression model. Because of this model, we will be able to detect

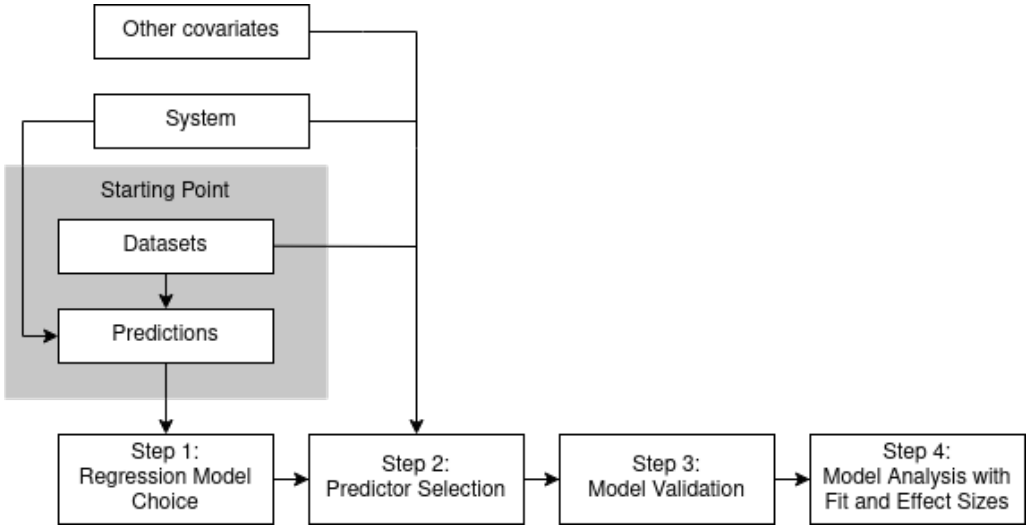

Figure 9: Evaluation procedure. Adapted from Dayanik et al. (2022).

potential biases, their influence, their significance, and their contribution to the total variance in the model. This presupposes a well-defined and annotated set of predictor variables, which rely on domain expertise and an extensively annotated corpus.

This evaluation procedure is illustrated in Figure 9 and will be explained step by step in the following sections.

4.3.1 Regression Model

As pointed out in Dayanik et al. (2022), classification tasks require a logistic regression model to analyze the output. As this is the case for this set of experiments, we used a basic logistic regression model as introduced in Section 2.6.

4.3.2 Predictor Selection

In our case, the predictor selection is given by the experiment structure. The TRUSTPILOT corpus makes both age and gender available to us, which will be used as predictors, as they are the main variables to be investigated. While we also use gender

L1	Danish vs. Mean	French vs. Mean
Danish	1	0
French	0	1
German	-1	-1

Table 2: Deviation encoding of the L1s in the TRUSTPILOT corpus

as a predictor for the ICLE corpus, we cannot use the age variable because of the unimodal age distribution in the corpus, which makes the age variable not yield any additional information.

Age is represented as its integer value, while gender is converted to 1 for female and 0 for male. We chose to encode the L1 using deviation regression coding, this way we can compare the performance of each L1 to the grand mean of all L1s.

Deviation coding works by comparing "the mean of the dependent variable for a given level to the grand mean of the dependent variable" (Chen et al., 2011). How such an encoding may look using the L1s from the TRUSTPILOT corpus is shown in Table 2. One variable is chosen as the "base level" and $n - 1$ new variables are created for the n variables given. The resulting values can be interpreted as the difference between the level and the mean. TRUSTPILOT then uses 2 new variables, shown in Table 2, while ICLE uses 15 new variables, as it has 16 L1s.

In Experiment 2 we also include whether the classified document originated from the training or the testing corpus, represented as a binary 1 or 0 variable. Since only two corpora are considered, but up to three exist in practice, it will always be mentioned which corpus is being referred to by the variable.

Finally, the variable we want to predict, is whether the classifier correctly processed and classified that document. This way, we can model the influence of each predictor has on the performance of the classifier.

This results in the following model for the TRUSTPILOT corpus:

$$p(\text{Correct}) \sim \sigma(\text{Age} + \text{Gender} + \text{Paraphrased} + \text{L1 Danish vs. Mean} + \text{L1 French vs. Mean})$$

And this larger model for the ICLE corpus:

$$\begin{aligned} p(\text{Correct}) \sim \sigma(&\text{Gender} + \text{Paraphrased} + \text{L1 Bulgarian vs. Mean} + \text{L1 Chinese vs. Mean} \\ &+ \text{L1 Czech vs. Mean} + \text{L1 Dutch vs. Mean} + \text{L1 Finnish vs. Mean} \\ &+ \text{L1 French vs. Mean} + \text{L1 German vs. Mean} + \text{L1 Italian vs. Mean} \\ &+ \text{L1 Japanese vs. Mean} + \text{L1 Norwegian vs. Mean} \\ &+ \text{L1 Polish vs. Mean} + \text{L1 Russian vs. Mean} + \text{L1 Spanish vs. Mean} \\ &+ \text{L1 Swedish vs. Mean} + \text{L1 Turkish vs. Mean}) \end{aligned}$$

4.3.3 Model Validation

According to Thompson et al. (2017) multicollinearity is the presence of "high levels of interdependence among predictors in a regression model", and can cause estimated coefficients to lose statistical significance or be assigned the wrong sign. To avoid this, we can measure multicollinearity using the variance inflation factor (VIF) as such:

$$VIF_i = \frac{1}{1 - R_i^2},$$

where R_i^2 is the correlation coefficient obtained when predicting α_i from all other available predictors, therefor the higher the VIF value is, the more multicollinearity is present between variables.

Tables 3 and 4 show the variance inflation factors (VIF) for each independent variable for each corpus and classifier. We chose a VIF cutoff value of 4, as higher values are considered to be problematic according to Thompson et al. (2017) because they indicate that the variable correlates strongly with another set of variables.

None of the VIF values for any of the two corpora rise above the set threshold of 4, so no additional investigation is necessary.

Gender	2.44
Paraphrased	1.72
L1	2.28

Table 3: VIF ICLE

Age	3.68
Gender	1.34
Paraphrased	1.88
L1	3.43

Table 4: VIF TP

4.3.4 Model Fit and Effect Sizes

To retrieve significant measures about the regression model, we not only need the coefficients of the model itself, but also additional measures to assess the quality of the model, and the contribution of each variable.

First, the goodness of fit is measured, while this is usually achieved using a R^2 measure, logistic regression models do not provide an equivalent measure. To circumvent this issue, a pseudo R^2 measure is used. Many measures have been proposed in the past, due to simplicity reasons we will use the McFadden pseudo R^2 measure, to guarantee comparable results regarding other values produced by all models.

Furthermore, we need to assess the relative importance of each variable, which is also not trivial for logistic regression models. We used Dominance Analysis (DA) (Budescu, 1993) to investigate this metric. DA estimates the contribution of each predictor variable to the explained variance in the outcome variable. This is done by applying regression to all possible subsets of predictor variables, and estimating the R^2 value for each regression. A variable is said to dominate another variable, if it is more useful than the other variable in all subset regressions.

This leaves us with the following metrics for each classifier-corpus pairing: The F1 score, a coefficient retrieved from the logistic regression model for each predictor, along with its significance level, the total dominance score calculated using DA for each predictor, the pseudo-R^2 value of the model, and the accuracy of the model.

Section 2.10 introduced the F1-Score, although a high value is generally an indicator of good performance, we mainly focus on the difference of F1-Scores between different setups.

The coefficients for the demographic properties are an indication of contribution to the performance of the outcome variable. Negative values indicate that when the value of that variable increases, the value of the outcome variable decreases. Positive values then indicate that when the value of the same variable increases, the value of the outcome variable increases. The coefficients of the L1s given by the logistic regression model as explained in 4.3.2 are a comparison of each L1 to the mean.

The significance level for each coefficient retrieved from the regression model indicates whether the predictor variable has a statistically significant relationship to the outcome variable. Generally, a higher significance level allows one to make better statements about the variable's influence. In all the following tables, we use * for $\alpha \leq 0.05$, ** for $\alpha \leq 0.01$, and *** for $\alpha \leq 0.001$.

A higher total dominance for a predictor variable from dominance analysis indicates that it has a stronger overall impact on the outcome variable relative to other predictor variables. Therefor, predictor variables with a higher total dominance are considered more important than other predictor variables.

The regression model also makes the pseudo-R^2, and the accuracy available, which are an indication of model fit. When our pseudo-R^2 measure increases, we know that a higher proportion of the dependent variable is explained by our independent variables, which is desirable.

5 Experimental Setup

5.1 Paraphrasing using GPT-3

We will create the new paraphrased versions of using GPT-3. GPT-3, which was already introduced in 2.9, offers different trained language models to process text. The different models have different prices and qualities, but the only model capable of paraphrasing a sentence and keeping its original meaning reliably is `text-davinci`. It is priced at $0.02 per 1000 tokens, where 1000 tokens correspond to approximately 750 words. OpenAI also provides an API which allows for automation of the paraphrasing.

The TRUSTPILOT corpus was paraphrased sentence-wise, as the documents are not long enough that any context has to be retained between sentences. The prompt given was: `"Paraphrase this to sound fluent:"` followed by the sentence itself. Parameters were set in a way that results are as deterministic as possible. Mainly, the `temperature` parameter was set to 0, which minimizes randomness and therefore maximizes reproducibility. The resulting corpus will be called TRUSTPILOT_P.

At first, the ICLE corpus was paraphrased the same way, sentence by sentence to create ICLE_S. This, however, is not optimal, as the context of surrounding sentences is not considered during paraphrasing. This means that the sentence being paraphrased does not consider references to surrounding sentences, nor does it process the already paraphrased versions of those sentences. Since the ICLE corpus contains learner essays with longer, coherent passages of text, which are longer than the user reviews in the TRUSTPILOT corpus, context from surrounding sentences is of much higher importance.

To combat this effect, a new method was used, its requirements were to include context from surrounding sentences, and to be reasonably efficient, since paraphrasing relies on the commercial OpenAI API.

The method is illustrated in Figure 10. To let the contents of surrounding sentences influence the result, a document d of n sentences S_i^j is paraphrased in groups

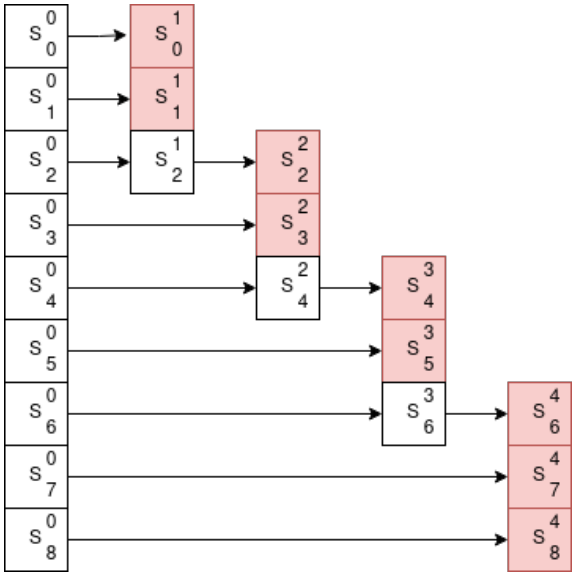

Figure 10: Paraphrasing procedure to let the context of surrounding sentences influence the final document

of three, where i indicates the sentence in the document, and j the amount of paraphrasing steps performed on that sentence. The first paraphrasing step ($i = 0$) simply paraphrases the first three sentences S_0^0, S_1^0 and S_2^0 to produce S_0^1, S_1^1 and S_2^1. Similarly to the other method, the prompt was: `"Paraphrase` S_i S_{i+1} S_{i+2} `to sound more fluent."`. Subsequent steps work by paraphrasing sentences S_i^{j-1}, S_{i+1}^0 and S_{i+2}^0, this way, the already paraphrased content from previous steps is considered. This is done until every sentence has been paraphrased once at the least, which takes $\lceil \frac{n}{2} \rceil$ steps. Each version of the sentence with the most paraphrasing steps is then merged with the others to create the final product, consisting of $\{S_i^{\lceil \frac{i}{2} \rceil} : 0 \leq i \leq |d|\}$. The new corpus will be referenced as ICLE_P.

The model used for the first ICLE paraphrasing was `text-davinci-002` as it was the newest at the time, all following paraphrasing procedures were performed using the newer model `text-davinci-003`.

Table 5 shows three examples of paraphrased sentences. Since the sentences were paraphrased for fluency only, no major restructuring has taken place. But

#	Original	Paraphrased
1	According to me the highest financial reward should get miners, pilots and sailors.	In my view, miners, pilots and sailors should be the ones who receive the highest financial compensation.
2	There are also some very good books written in our modern world.	Additionally, there are some excellent books written in our contemporary society.
3	A statement, such as the quote from George Orwell's novel Animal Farm "all animals are equal, but some animals are more equal than others", can be studied from a couple of points of view and al!	The quote from George Orwell's Animal Farm, "all animals are equal, but some animals are more equal than others", can be interpreted in various ways, all of which are relevant to our current times.

Table 5: Paraphrasing results by GPT-3

common stylistic errors, such as `"very good books"` were not only replaced by better sounding combinations, such as `"excellent books"` according to context, but also integrated into more fluent wordings. Moreover, important to this process, is that quotes do not get paraphrased, which they do not, as can be seen in the third example, this indicates that a quote is treated correctly according to context.

Since paraphrasing a text will likely change its structure, we also briefly investigate how the original corpora differ from the newly created paraphrased versions.

Table 6 shows some basic characteristics of all the corpora mentioned. Values worth mentioning are highlighted and will be discussed in the following section.

Although the average number of words in a document is a very general metric, it does vary noticeably for the ICLE corpora. Documents from ICLE_P are on average 45 words longer than the originals, this is expected, since GPT-3's language skills and vocabulary stem from a collection of documents of proficient English, which

	ICLE	ICLE_P	ICLE_S	TP	TP_P
Average # of Words per Document	**740.19**	**785**	**705.23**	**54.02**	**50.26**
Average # of Sentences per Document	33.68	33.33	**37**	3.03	3.02
Average # of Words per Sentence	**20.72**	**22.35**	**18.11**	**14.92**	**14.71**
Average Length of Word	4.24	**4.37**	4.25	3.99	3.92
# of different words	**68558**	43608	48332	**16172**	10900

Table 6: Corpus analysis after paraphrasing (TRUSTPILOT has been abbreviated as TP)

generates more elaborated wordings. ICLE_S however, is shorter on average, this is not expected, as it used the same language model as ICLE_P. It is likely, however, that the sentences were shortened for stylistic purposes during paraphrasing, and because no surrounding context was present, no references could be made to other content, resulting in shorter sentences. This corresponds with the average amount of words per sentence, which is higher for ICLE_P and lower for ICLE_S, and is also illustrated in Figure 11. In the distribution plot, it is clearly visible that ICLE_P has a similar distribution to the original ICLE corpus, but its sentences are longer on average. The ICLE_S corpus' distribution shows a much larger peak at 15 words per sentence, but shows a sharper drop than the other corpora, meaning that long sentences were shortened, and sentences that were short already, were kept at a similar length.

The average length of a word does not show any noticeable changes, only ICLE_P has slightly longer words than the other two ICLE corpora, although only by approximately 3%.

An expected phenomenon, is that the number of unique words in the paraphrased corpora is significantly lower. This is because the original corpora contain spelling

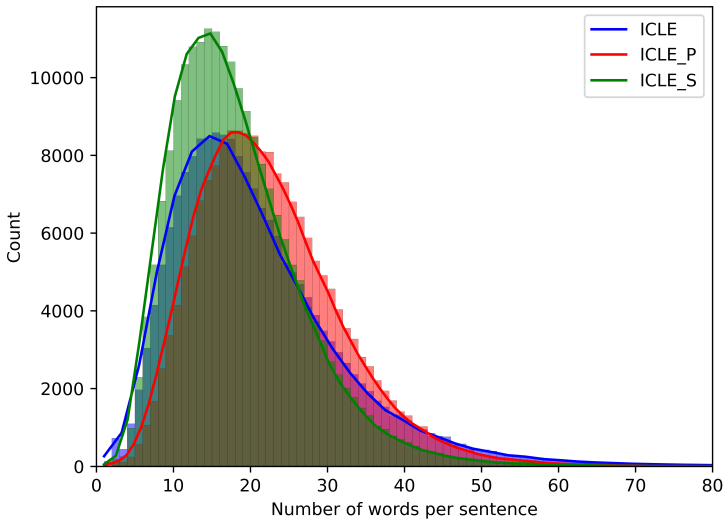

Figure 11: Distribution of number of words in a sentence in the ICLE corpora

errors, something the paraphrased versions do not, since the GPT-3 language model will attempt to correct them.

The TRUSTPILOT corpora show similar phenomena, the average number of words per document has fallen in TRUSTPILOT_P similarly to ICLE_S. Figure 12 shows, that the distribution change from TRUSTPILOT to TRUSTPILOT_P resembles the change between ICLE and ICLE_P, in the way that all sentences have become longer overall.

The vocabulary size of TRUSTPILOT_P has also shrunk to 67.4% of the original size, for the same reasons present in the ICLE corpora.

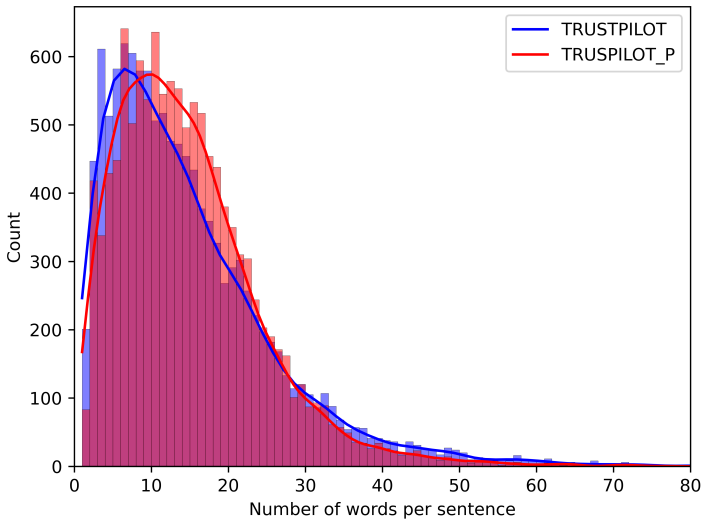

Figure 12: Distribution of number of words in a sentence in the TRUSTPILOT corpora

5.2 SVM

The SVM was implemented using the `scikit-learn`, `numpy` and `nltk` libraries in Python. First, the data was filtered and prepared to fit the structure and criteria mentioned in Section 3. Afterward the TF-IDF matrix was constructed, to achieve that, each document was tokenized and then converted into uni- and bigrams.

To retrieve results, the SVM was used to classify the documents of the testing set of the current training-testing pair. Performance was measured using a weighted F1-Score, which calculates the macro F1-Score for each class, and then finds their average weighted by the number of true instances per class. This was done once for the entire testing set, and then for each predefined demographic group. The weighted F1-score was chosen to counteract the class distribution, since the classes in all the datasets are imbalanced.

5.3 Language Model

Although the BERT model uses the same data pairs as the SVM, its ability to process raw text instead of vectors meant that the documents only had to be tokenized using the matching tokenizer for the BERT model. The libraries needed for the BERT classifier and the tokenizer were `transformers` and `pytorch`. The pretrained model used for this task was `bert-base-uncased`, which meant that the documents were tokenized using the tokenizer with the same name.

The 512 token limit of the tokenizer is often reached by the ICLE corpus, as can be seen in Figure 3, many of the available methods to circumvent this limit are not feasible in the scope of this work. Alternatives to truncation of the document to 512 tokens include using a different tokenizer, such as *Longformer* by Beltagy et al. (2020) or splitting documents of more than 512 tokens into smaller pieces to classify each of them separately. The latter option, however, raises many more issues and design ideas, that do not match the scope of this work, so truncation was used anyway. Although both Sun et al. (2019a) and Mutasodirin and Prasojo (2021) mention potential improvements using special truncation techniques, we do not consider them relevant to the scope of this work, as we are mainly interested in performance differences instead of raw performance. Therefor, documents longer than 512 tokens are automatically truncated in our setup.

After testing, the hyperparameters for the model were chosen as recommended by Devlin et al. (2019), as they achieved the best performance, even in the restricted number of epochs. The batch size was set to 8 as this was the highest reasonable number the GPU (NVIDIA RTX 3080 10GB) used could handle. We used the Adam optimizer (Kingma and Ba, 2014) in training and set its learning rate to $3 * 10^{-5}$. Finally, the model was limited to 4 epochs of training, as any further epochs raised the accuracy on the training set, while maintaining the accuracy of the testing set. This indicates an increased risk of overfitting to features in the training set, which are not relevant in other documents.

Finally, the performance metrics were chosen to be the same as for the SVM, to retain comparability.

5.4 Evaluation

The regression analysis, which includes the VIF analysis, was implemented using the `statsmodels` library. The DA was performed using the `dominance-analysis` Python library, which also uses `statsmodels` to calculate all necessary logistic regression. Since we use the same library for all logistic regressions, we can assume all values, such as the previously mentioned McFadden pseudo-R^2 measure, to be comparable between each other, as the underlying system remains constant.

6 Results

In this section, we will discuss the results of both experiments and compare the different setups. Since both experiments generate numerous results, the tables containing the results have been reduced to the values required to show the main points in these sections. The full tables containing all values can be found in the appendix.

6.1 Experiment 1

6.1.1 ICLE

	SVM			BERT		
	ICLE	ICLE_P	ICLE_S	ICLE	ICLE_P	ICLE_S
Overall	83.75	71.87	74.60	75.79	64.79	69.54
Male	84.35	73.44	73.69	78.65	66.35	72.32
Female	83.72	71.49	75.07	74.88	64.44	68.97
Bias	0.99	0.97	1.02	0.95	0.97	0.95

Table 7: F1-Scores for the ICLE corpora (Bias = $F1_{Female}/F1_{Male}$)

Table 7 shows the F1-scores for the ICLE corpus. The F1-scores have been split up by gender to allow for the analysis of the gender bias.

Our hypothesis was that the performance of classifiers would decrease significantly on paraphrased corpora. Additionally, we expected some level of bias when investigating the demographic properties gender and age.

The first observable phenomenon is the clear drop in F1-score in all paraphrased versions of the original corpora. ICLE_P is the corpus showing the strongest decrease in F1-score, lowering the score by over 10 points compared to the original corpus. However, ICLE_S also shows similar effects regardless of classifier, indicating that

it applies to all ICLE corpora. The clear decrease in F1-score across both classifiers suggests that paraphrasing a corpus does indeed lower the performance when trying to deduce the L1 of the document's author.

Although this alone supports the main hypothesis for this experiment, multiple other phenomena can be observed from these results.

At first, it is noticeable that although the ICLE_S corpus performs worse than the unmodified ICLE corpus, it does perform better than the thoroughly paraphrased ICLE_P corpus. This is easily explained by looking at the way the corpora were paraphrased. Since ICLE_S was paraphrased sentence by sentence, it is very similar to the original corpus. The overall structure was not changed, as each sentence was only replaced by a modified version of itself. This also restricted the freedom the GPT model had to restructure the sentence, as it did not have any context available.

When inspecting gender bias, a small and consistent male bias can be seen. Only one of the six corpora show a slight female bias, which is also small, while the ICLE and ICLE_S corpora with the BERT model showing the highest bias at a ratio of 0.95. Although some bias is present, this does not match the expected extent.

The SVM performs better on the ICLE corpora, which can be explained by the length of the documents in the corpora. Many documents supersede the 512 token limit of the BERT tokenizer mentioned in Section 2.8, as can be seen in Table 3. When the BERT language model trains on these documents, much of the documents is discarded, a problem which the SVM does not have.

Although the age of the authors was also investigated, it did not yield any substantial results and has been left out, as performance was highly dictated by data availability.

The results of the regression analysis have been displayed in Table 8. The table has been severely reduced to the essential results relevant to the analysis. The full version is available in the appendix.

The first table shows the results of the regression analysis of the original ICLE corpus, since the results were similar for all three versions of the corpus. Both language-independent factors stand out because they are not considered important

		Length	Gender	BG	CN	DU	FI	FR	IT	JP	NO	SE	R^2	Acc.
ICLE														
SVM	Coef.	0.00	0.33	-1.11***	1.37***	-0.21	-0.68	-1.02***	-0.21	1.61*	-0.64*	-0.56*	0.09	0.84
	DA	0.00	0.00	0.02	0.02	0.00	0.00	0.01	0.00	0.01	0.01	0.00		
BERT	Coef.	0.00	0.22	0.13	3.03***	-1.34***	-1.32***	-1.24***	0.81*	0.99*	-0.72*	-1.13***	0.18	0.77
	DA	0.01	0.00	0.00	0.04	0.02	0.02	0.03	0.00	0.00	0.01	0.02		

Table 8: Results from regression model and DA for ICLE corpora

by the DA. In addition, the coefficients are not significant and 0 for the length predictor. The gender predictor, although insignificant, is slightly positive, indicating a stronger positive influence by documents with female authors.

When looking at the L1 predictors, some L1s are better predictors for correct classification than others. The ones that stand out as good predictors are Chinese and Japanese. Chinese is by far the strongest predictor, showing the highest coefficient and the highest importance by the DA, along with a high significance level. That Chinese and Japanese are the two L1s standing out as good predictors can be explained by the fact that they are the only two languages, apart from Tswana, that are not Indo-European. Therefore, the documents with Chinese and Japanese authors present more unique features, that can be detected by both classifiers.

Most of the other language predictors stand out due to the negative sign, along with a high significance level. Especially French stands out, as it also has a high importance assigned by the DA. This may be explained by French being closely linked to English in terms of historical language development due to geographic and cultural reasons.

Although other L1s such as Bulgarian and Italian show noticeable behavior, we cannot reasonably explain their behavior without speculating.

Finally, the R^2 measure is significantly higher when using the BERT model, this is true for all three ICLE corpora. This means that a larger proportion of the

variance in the dependent variable can be explained by the predictor variables when using the BERT model. This is in line with the proportion of coefficients in each classifier that have a high significant level.

6.1.2 Trustpilot

	SVM		BERT	
	TRUSTPILOT	TRUSTPILOT_P	TRUSTPILOT	TRUSTPILOT_P
Overall	72.58	65.30	77.30	70.36
Male	72.93	66.27	76.94	70.57
Female	71.41	62.35	78.24	69.82
Bias	0.98	0.94	1.02	0.99

Table 9: F1-Scores for the TRUSTPILOT corpora (Bias = $F1_{Female}/F1_{Male}$)

Table 9 shows the F1-scores for the TRUSTPILOT corpora. They have also been split up by gender.

The results show a very similar behavior to the ICLE corpora, where paraphrasing a corpus entails a significant decrease in performance regardless of the classifier used. Both corpora show a similar amount of gender bias to the ICLE corpora, where three out of four classifiers were male-biased, although none surpassed the ratio of 0.98.

In this case, the BERT model outperforms the SVM, but this has the same explanation as the SVM outperforming the BERT model on ICLE corpora. The majority of documents in the TRUSTPILOT corpora do not exceed the 512 token limit of the tokenizer. Since BERT produces higher F1-scores on arguably equal terms, we can assume that the model outperforms the SVM on this corpus.

Due to the same reasons described in Section 6.1.1, age has been left out and all statements about the influence of age on classification has to stem from the following

		Length	Gender	Age	DK	FR	R^2	Acc.
TRUSTPILOT								
SVM	Coef.	0.00*	-0.02	0.00	0.5***	0.02	0.03	0.72
	DA	0.01	0.00	0.00	0.02	0.00		
BERT	Coef.	0.00	0.12	0.01	0.59***	0.07	0.05	0.78
	DA	0.00	0.00	0.00	0.04	0.01		
TRUSTPILOT_P								
SVM	Coef.	0.00	-0.15	0.00	0.6***	-0.24*	0.04	0.65
	DA	0.00	0.00	0.00	0.03	0.00		
BERT	Coef.	0.00	-0.04	0.01	0.61***	-0.16	0.04	0.71
	DA	0.00	0.00	0.00	0.03	0.00		

Table 10: Results from regression model and DA for TRUSTPILOT corpora

analysis of the regression model.

The results of the regression analysis on the TRUSTPILOT corpora is shown in Table 10.

The results of the analysis show similar traits to the analysis of the ICLE corpora, with some interesting differences.

Both coefficients for length and gender stay close to 0, and their importance by the DA is 0 as well. This analysis also includes age as a predictor, which again shows no importance or significant coefficient. Now that all demographic variables seem to be unimportant, we can discard our starting assumption that demographic properties would be strong indicators of classifier performance.

Since German has been selected as the "base level" for the deviation coding, only Danish and French remain as L1 predictors. It is clear that Danish is noticeably easier to classify than German, as all coefficients are above 0.5 and highly significant. The DA marks this predictor as highly important, with values between 0.02 and 0.04, while no other predictor rises above 0.01. This is likely attributed to

the data distribution, which strongly favors documents with a Danish L1. Another plausible explanation is that Danish is structurally different from English because Danish natives tend to construct subordinate clauses using "main clause" word order (Christensen et al., 2020), something both classifiers may detect easily.

Another interesting phenomenon is that the coefficients for the French predictor are slightly lower for the TRUSTPILOT_P than for the original corpus. This can be interpreted as French being harder to classify than German after being paraphrased, speaking in relative terms. On average, it can be said, that documents by French authors are masked better by the paraphrasing procedure, as the performance improves relatively to other L1s after paraphrasing.

Lastly, the R^2 value is now almost equal between the two classifiers and the two corpora, as only very few documents cross the 512 token threshold.

6.2 Experiment 2

This experiment provides similar results to the first experiment, with the important distinction, that the classifiers have classified a different corpus in addition to the one they were trained on. The notation is explained here for easier understanding in the following sections.

The corpus pairs consist of two corpora, `Corpus 1` and `Corpus 2`, and are denoted as `Corpus 1 - Corpus 2`. `Corpus 1` is the testing dataset on which the classifiers trained on. The testing set is the union of both corpora `Corpus 1` and `Corpus 2`.

When a corpus pair is mentioned as "ICLE - ICLE_P", it means that the classifier trained on the ICLE corpus, and was used to classify documents from both ICLE and ICLE_P.

The focus of this experiment lies on the domain transfer effects a classifier may show. Although the corpus pairs stem from the same original corpus, we argue that they are different enough, that these effects may be considered effects of domain transfer.

6.2.1 ICLE

	SVM						BERT					
	ICLE - ICLE_P	ICLE - ICLE_S	ICLE_P - ICLE	ICLE_P - ICLE_S	ICLE_S - ICLE	ICLE_S - ICLE_P	ICLE - ICLE_P	ICLE - ICLE_S	ICLE_P - ICLE	ICLE_P - ICLE_S	ICLE_S - ICLE	ICLE_S - ICLE_P
Overall	66.80	75.80	71.58	69.66	78.63	68.22	60.59	63.67	67.00	66.39	69.58	64.74
Male	67.92	74.64	73.90	71.82	78.57	70.65	62.72	66.60	69.85	68.23	72.17	67.11
Female	66.48	76.23	70.99	69.12	78.84	67.52	60.04	62.88	66.38	66.04	69.17	64.24
Bias	0.98	1.02	0.96	0.96	1.00	0.96	0.96	0.94	0.95	0.97	0.96	0.96

Table 11: F1-Scores for the ICLE corpora (Bias = $F1_{Female}/F1_{Male}$)

The hypothesis for this experiment was similar to that of the first experiment, in that we expected performance of classifiers to decrease significantly on paraphrased corpora. Due to the added aspect of the domain transfer in this experiment, we also expected the performance to decrease more when classifying a paraphrased corpus compared to when we classify the original corpus when we trained on a paraphrased corpus. Furthermore, we assumed to find similar biases in the demographic factors.

Similarly to Experiment 1 in Section 6.1.1, Table 11 shows the overall F1-scores, as well as the F1-scores split by gender, for all ICLE corpus pairs. The performance discrepancies between the SVM and the BERT model behave similarly for the same reasons mentioned before. Even the gender bias stays similarly male-biased in 10 of the 12 corpus-pairs.

When inspecting the F1-scores for the pair tested, some interesting patterns arise. Classifiers which have been trained on the unaltered ICLE corpus and have been applied to a different version of the corpus perform significantly worse, when

		Paraphrased	Length	Gender	CZ	ES	FI	PL	R^2	Acc.
ICLE - ICLE_P										
SVM	Coef.	0.89***	0.00**	0.2	0.07	-0.3	-0.65*	-1.04***	0.15	0.78
	DA	0.03	0.01	0.00	0.00	0.00	0.01	0.02		
BERT	Coef.	0.49***	0.00**	0.17	-0.37	-0.21	-1.68***	-1.55***	0.25	0.75
	DA	0.01	0.01	0.00	0.00	0.00	0.03	0.05		
ICLE_P - ICLE										
SVM	Coef.	-0.02	0.00	0.2	-0.77***	0.06	-0.32	0.04	0.10	0.72
	DA	0.00	0.00	0.00	0.01	0.00	0.00	0.00		
BERT	Coef.	0.14	0.00	0.27*	-0.83***	-0.93***	-0.67**	-0.03	0.18	0.72
	DA	0.00	0.01	0.00	0.01	0.01	0.01	0.00		

Table 12: Results from regression model and DA for ICLE corpora

compared to the same classifier applied to the original corpus. A notable exception is the ICLE-ICLE_S pairing, as it achieved a higher F1-score than the classifier that trained and classified ICLE_S. This can be attributed to the similarity of ICLE_S to the original corpus, although this does not explain the increase in performance itself.

Classifiers trained on "harder" corpora, such as ICLE_P and ICLE_S, do not necessarily perform worse, however, as some combinations yielded better results than their counterparts that classified the same corpus they were trained on.

When looking at the results of the regression analysis, the most interesting predictor is the "Paraphrased" variable, as it will indicate whether classifying a document from a different corpus is a good indicator for performance.

When looking at the analysis results of the ICLE corpora from Table 12, we notice multiple similar phenomena that have already been discussed in the first experiment. Mainly, the R^2 measure, which is higher when using the BERT classifier, and the

demographic predictors, which show no influence of the variables apart from a slight positive female influence, have not changed.

The new "Paraphrased" predictor does confirm our hypothesis, when a classifier is trained on the original ICLE corpus, it performs much better when classifying a document from the same corpus, instead of one from the ICLE_P corpus. This in itself is expected, as domain transfer always increases difficulty. However, when we look at the same values in the pairing, in which the classifier was trained on ICLE_P, we notice that none of the two corpora were easier to classify than the other, as the coefficients are low, have no significance level, and the importance assigned by the DA is 0.00. This indicates that a classifier that has been trained on "harder" paraphrased corpora does not gain any performance advantage from classifying the original corpus.

The L1 predictors in the Table have been chosen by whether they differ significantly from the results of Experiment 1, or if their values changed significantly between corpus-pairing in this experiment.

The predictor for Czech, indicates that it is harder to classify (compared to Tswana) when the classifier was trained on a corpus different from the original ICLE version, suggesting that Czech is masked better than some other languages when training on paraphrased corpora. The same behavior can be observed for documents with Spanish as a L1, but only when using the BERT classifier, the SVM does not change the relative difficulty.

When classifying documents with a Finnish L1, it appears as if training on the paraphrased corpus makes the documents easier to classify than before. Polish shows a similar change, however, the predictor loses its significance levels, and its importance.

6.2.2 Trustpilot

	SVM		BERT	
	TP - TP_P	TP_P - TP	TP - TP_P	TP_P - TP
Overall	61.24	68.84	65.65	71.92
Male	60.97	69.55	66.20	71.90
Female	62.10	66.50	63.76	71.73
Bias	1.02	0.96	0.96	1.00

Table 13: F1-Scores for the TRUSTPILOT corpora (Bias = $F1_{Female}/F1_{Male}$) (TRUSTPILOT abbreviated as TP)

The F1-scores of each corpus pair and classifier are shown in Table 13, along with the split by gender. BERT outperforms the SVM once again, although the reasons remain the same as in Section 6.1.2. The gender bias remains low but male-biased in three out of four results.

Although the same patterns arise in the TRUSTPILOT corpora when inspecting performance differences among corpus pairs, they are easier to display. The effect of a classifier performing better when trained on a "harder" corpus is stronger. The TRUSTPILOT_P - TRUSTPILOT combination is a good example of this effect, as it achieved a F1-score of 71.92, while the classifier that classified TRUSTPILOT_P instead of only being trained on it, achieved a lower F1-score of 70.36.

This behavior can be expressed more easily when labelling corpora as "harder" or "easier" than others, based on whether they were paraphrased. When doing so, a clear performance hierarchy can be constructed, which is shown as an example using the TRUSTPILOT classifiers in Table 14.

F1-score	TP	>	TP_P - TP	>	TP_P	>	TP - TP_P
SVM	72.58	>	68.84	>	65.30	>	61.24
BERT	77.30	>	71.92	>	70.36	>	65.65

Table 14: Sorted F1-scores of all classifiers using the TRUSTPILOT corpora

The classifier training on the easier corpus and classifies it, performs the best. The next-best classifier is the one training on harder data, but classifies the easier corpus. After that comes the classifier working solely on the harder corpus, and the worst performing classifier then is the one, that trains on the easy corpus, but then classifies the hard corpus.

The same structure can be built for the ICLE corpora, when declaring the ICLE_S corpus to be a middle-ground between ICLE and ICLE_P. In this case, some exceptions do happen, but the general idea still holds.

The regression analysis for the Trustpilot corpora pairs, shown in Table 15, stays mostly identical to the one shown in Section 6.1.2. The demographic predictors, the language predictors, and the R^2 measure all present identical behavior to before.

However, similarly to the analysis for the ICLE corpora, the "Paraphrased" predictor is of interest. It also shows that classifying documents from a paraphrased corpus is harder, than classifying documents from the original and unmodified corpus the classifier was trained on. Again, when using the paraphrased corpus to train, this effect does not occur anymore, and the classifier does not perform significantly better or worse on documents from one corpus or another.

		Paraphrased	Length	Gender	Age	DK	FR	R^2	Acc.
TRUSTPILOT - TRUSTPILOT_P									
SVM	Coef.	0.45***	0.00	0.05	0.00	0.69***	-0.13	0.06	0.68
	DA	0.01	0.00	0.00	0.00	0.05	0.00		
BERT	Coef.	0.54***	0.00	0.01	0.00	0.69***	0.09	0.08	0.74
	DA	0.01	0.00	0.00	0.00	0.05	0.01		
TRUSTPILOT_P - TRUSTPILOT									
SVM	Coef.	0.18	0.00*	-0.13	0.00	0.51***	-0.18*	0.03	0.67
	DA	0.00	0.00	0.00	0.00	0.02	0.00		
BERT	Coef.	0.07	0.00	0.01	0.01**	0.55***	-0.15	0.03	0.72
	DA	0.00	0.00	0.00	0.01	0.03	0.00		

Table 15: Results from regression model and DA for TRUSTPILOT corpora

7 Discussion

Experiment 1 shows that NLI classification performance decreases significantly when using paraphrased versions of the same corpus. It has also shown that the method used to paraphrase a document has an impact on the performance. When inspecting the F1-scores, we only detected a small male bias, which is lower than typical for similar experiments (Dayanik et al., 2022). In general, performance measured using an F1-score was dictated by data availability, which made analyzing the F1-score by age obsolete, as its distribution was unimodal and concentrated. Both the low impact of gender, and the low impact of age did not match our assumptions, as we expected there to be a significant influence. But both cases are hugely influenced by data distribution, which may hide some larger biases.

The effects of truncation by the BERT tokenizer can be observed in the results, as it performs worse than the SVM on the ICLE corpora, which contain longer

documents on average. This also affected the R^2 values of the regression analysis, which are higher than the SVMs when the BERT language model only had truncated data available. Another plausible explanation is that misspellings of words, which may be motivated by the L1, are tokenized in such a differing way from the correctly spelled token that they influence the performance of the language model. This can be mitigated in many ways, but a word-based spelling correction so that the sentence structure remains equal is likely to reduce the effect while preserving our other findings.

Experiment 2 confirmed the findings of the first experiment, and additionally allowed for more findings. Together with the F1-scores available from the first experiment, each classifier can be compared to create a hierarchy in performance. This hierarchy indicates that the classifiers can be sorted from the classifier that only classifies the original corpus, to the classifier that trains on the original corpus, and additionally classifies the paraphrased version.

The regression analysis of both experiments show that, as seen in the F1-scores, demographic predictors, as well as the length predictor did not contribute significantly to any classifier. This finding is not expected, as our hypothesis assumed a stronger influence of demographic factors on the classifier's performance. We cannot give a reasonable explanation for this behavior. However, we can assume that the corpora used suffer from topic bias and other demographic biases as can be seen in other studies (Brooke and Hirst (2013), Hovy et al. (2015), Hovy (2015)). This indicates that the effect of lack of influence from demographic factors is likely introduced during this work.

The important finding from the regression analysis of Experiment 2 is, that when trained on an original corpus, the classifier performs best when classifying documents from the same corpus. When trained on an altered version of the corpus, however, this phenomenon fades, as no classifier performed better or worse on documents from the original or from the altered corpus.

Many of the language predictors in both experiments were not significant or important enough to be mentioned, and some mentioned do not offer an obvious

explanation. But we observed some L1s providing additional information and performance to the classifier in both experiments. Mainly, the Chinese and Japanese stood out as easy to classify, due to them not being Indo-European. Danish was also easy to classify, most likely due to the data availability or the subordinate clause word order typical for Danish natives (Christensen et al., 2020). Other L1s, such as French, Norwegian, and Swedish, presented themselves as harder to classify, although the reasons were often unclear.

Very few languages became noticeably harder or easier to classify after paraphrasing. An example of such an effect are documents of French natives, as they are slightly harder to classify after being paraphrased in the TRUSTPILOT corpus. This effect is explained by the fact that documents from French natives are more likely to present some spelling errors over others. Other authors of other native languages also present this effect, just with other spelling errors. Stehwien and Padó (2016) lists "exemple" as a common spelling mistake made by French authors. Although correctly spelled, "Indeed" and "To conclude" are two formal sentence introductions also often used by French authors. Occurrences of these words are likely to be an indicator for the classifier that are changed through paraphrasing.

Additionally to the just mentioned frequency of specific spelling errors by authors with different native languages, the ICLE corpus poses another explanation for easily identifiable tokens. As mentioned in Section 3.1, Brooke and Hirst (2013) demonstrated that the ICLE corpus suffers from significant topic bias. This changes the vocabulary used in each document, but it is difficult to attribute a change in performance through paraphrasing. Nevertheless, it is likely to affect the performance of the classifiers. We assume an increase in performance because of the topic bias, and we also assume this increase in performance to distribute unevenly between all L1s. However, it is difficult to quantify the influence of ICLE's topic bias on a per-L1 basis.

8 Conclusion

In this thesis, we utilized the ICLE corpus, a learner corpus, and the TRUSTPILOT corpus, a web-scraped corpus, to conduct NLI experiments. We created new versions of both corpora by paraphrasing each document using GPT-3. To better paraphrase the longer documents of the ICLE corpus, we developed a new technique to efficiently paraphrase a document, while allowing for structural changes within itself to create a second paraphrased version of the corpus. Using the five corpora, and additional demographic metadata such as age and gender, together with the length of each document, we performed two different NLI experiments. To classify each document, we used two different NLI classifiers, an SVM which was trained using TF-IDF constructed from uni- and bigrams, and a pre-trained BERT language. We analyzed our results using typical performance measures, such as the F1-score, and an advanced combination of regression analysis and dominance analysis.

We conducted two experiments in which we detected a significant performance decrease through the influence of the paraphrased corpora. In both experiments we noticed the lack of influence by demographic factors, while overall L1s were a better predictor for a correct classification.

In this work, we showed how performing NLI experiments are heavily influenced when the documents used to perform such experiments are paraphrased. This finding is in line with the expected results. We also expected demographic factors to have a significant influence on the performance of the NLI classifiers themselves, but found that the influence of such factors is negligibly small or non-existent. This contradicts our expectations, as well as findings by previous works such as Hovy (2015) and Sun et al. (2019b). We found the L1 of a document's author to be a better predictor of classification success. The regression analysis allowed us to quantify the relative influence of each L1, which in turn allowed us to detect performance differences in classification motivated by features unique to a language.

The data available, the execution of the experiments, and the resulting findings were limited by factors common for NLP experiments. The corpora choice was highly dictated by the availability of corpora containing the necessary metadata for our

experiments. This limits the amount of data available to us, which impacts the performance of classifiers which rely on training data. The quality of the data can also be questioned, since both corpora pose slight concerns when it comes to their quality. Brooke and Hirst (2013) already demonstrated that ICLE suffers from topic bias, which cannot be easily removed. The TRUSTPILOT corpus suffers from notable gender bias (Hovy, 2015), and additionally consists of reviews which tend to be short on average, as can be seen in Table 5.

The method we developed to paraphrase a document thoroughly to allow for the retention of surrounding context fulfilled its purpose. However, it was designed with its cost in mind, which is a limitation. More resources would allow for the design of an even more thorough paraphrasing method, which we assume would amplify the impact of paraphrasing on NLI tasks.

Although computing power was a limiting factor during the training of the BERT language model, it did not impact the results, as more computing power would only deliver the results faster. When using larger corpora with longer documents, however, the computing power might be insufficient to conduct the experiments in a reasonable time.

The findings in this work can be improved by using larger corpora, mostly because the BERT language model profits from a larger quantity of datapoints (Steinbakken and Gambäck, 2020). Additionally, the paraphrasing method developed in this work, can be improved. Although the method sufficed to prove our hypothesis, it is reasonable to assume that a more sophisticated and elaborate method could further "hide" a L1 from an NLI classifier.

We chose BERT as the language model to represent neural network language models, but many more exist and new ones are being developed. An analysis comparing the effect of paraphrasing on NLI, along with the influence of demographic factors when using different neural network language models is likely to yield further findings in this field.

Finally, our hypothesis that demographic metadata would be a significant predictor was shown to be untrue, however, the reason behind this behavior is unclear.

Further research could investigate whether this behavior is systematic or bound to our experimental setup.

9 Appendix

			ICLE	ICLE_P	ICLE_S
SVM	Male	Overall	84.35	73.44	73.69
	Male	10-30	85.32	74.32	74.49
	Male	31-99	52.14	33.33	57.50
	Female	Overall	83.72	71.49	75.07
	Female	10-30	84.17	68.07	75.40
	Female	31-99	76.70	68.07	70.47
	Overall		83.75	71.87	74.60
BERT	Male	Overall	78.65	66.35	72.32
	Male	10-30	79.44	68.01	73.83
	Male	31-99	65.48	21.43	45.00
	Female	Overall	74.88	64.44	68.97
	Female	10-30	75.11	64.77	69.30
	Female	31-99	65.29	59.13	62.72
	Overall		75.79	64.79	69.54

Table 16: F1-Scores for the ICLE corpora

			Trustpilot	Trustpilot_P
SVM	Male	Overall	72.93	66.27
	Male	10-30	68.25	60.38
	Male	31-55	74.10	67.27
	Male	56-99	75.33	75.94
	Female	Overall	71.41	62.35
	Female	10-30	62.02	60.88
	Female	31-55	81.04	68.06
	Female	56-99	64.55	47.75
	Overall		72.58	65.30
BERT	Male	Overall	76.94	70.57
	Male	10-30	69.55	63.40
	Male	31-55	80.85	73.97
	Male	56-99	83.74	78.58
	Female	Overall	78.24	69.82
	Female	10-30	76.04	69.20
	Female	31-55	82.10	71.55
	Female	56-99	79.39	66.44
	Overall		77.30	70.36

Table 17: F1-Scores for the Trustpilot corpora

		Length	Gender	BG	CN	CZ	DE	DU	ES	FI	FR	IT	JP	NO	PL	RU	SE	TR	R^2	Acc.
ICLE																				
SVM	Coef.	0.00	0.33	-1.11***	1.37***	-0.13	-0.21	-0.21	0.22	-0.68	-1.02***	-0.21	1.61*	-0.64*	-0.08	0.16	-0.56*	-0.32	0.09	0.84
	DA	0.00	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00		
BERT	Coef.	0.00	0.22	0.13	3.03***	-0.63*	-0.42	-1.34***	-0.46	-1.32***	-1.24***	0.81*	0.99*	-0.72*	-0.52	-0.07	-1.13***	0.71	0.18	0.77
	DA	0.01	0.00	0.00	0.04	0.01	0.00	0.02	0.00	0.02	0.03	0.00	0.00	0.01	0.00	0.00	0.02	0.00		
ICLE_P																				
SVM	Coef.	0.00	0.23	-0.98***	1.8***	-0.46	0.03	-0.24	-0.1	-0.44	-0.68*	0.75*	0.97**	-0.51	-0.33	-0.32	-0.24	-0.76***	0.11	0.73
	DA	0.00	0.00	0.02	0.03	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01		
BERT	Coef.	0.00	0.31	0.2	3.16***	-0.98**	0.08	-1.01**	-0.93**	-0.61*	-0.84**	0.84**	1.03**	-1.08***	-0.12	-0.6*	-0.7**	-0.06	0.18	0.72
	DA	0.01	0.00	0.00	0.07	0.01	0.00	0.01	0.01	0.01	0.01	0.00	0.01	0.02	0.00	0.01	0.01	0.00		
ICLE_S																				
SVM	Coef.	0.00	0.38*	-0.86***	1.62***	-0.72*	0.01	-0.69*	-0.08	-0.66	-0.49	0.31	1.33**	-0.42	0.21	-0.05	-0.12	-0.64**	0.09	0.75
	DA	0.00	0.00	0.02	0.03	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01		
BERT	Coef.	0.00	0.2	-0.14	2.81***	-0.18	-0.24	-1.16***	-0.65*	-0.99**	-0.79**	1.01**	1.13**	-0.74**	-0.59*	-0.35	-0.93***	0.49	0.15	0.72
	DA	0.01	0.00	0.00	0.05	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.00	0.02	0.00		

Table 18: Results from regression model and DA for ICLE corpora

			ICLE - ICLE_P	ICLE - ICLE_S	ICLE_P - ICLE	ICLE_P - ICLE_S	ICLE_S - ICLE	ICLE_S - ICLE_P
SVM	Male	Overall	67.92	74.64	73.90	71.82	78.57	70.65
	Male	10-30	68.79	75.35	75.09	73.23	79.28	72.10
	Male	31-99	33.93	52.14	32.14	37.50	52.14	27.50
	Female	Overall	66.48	76.23	70.99	69.12	78.84	67.52
	Female	10-30	66.67	76.37	71.34	69.07	79.25	67.88
	Female	31-99	75.72	80.29	63.15	77.58	69.64	62.28
	Overall		66.80	75.80	71.58	69.66	78.63	68.22
BERT	Male	Overall	62.72	66.60	69.85	68.23	72.17	67.11
	Male	10-30	63.96	68.02	71.91	69.98	73.49	68.46
	Male	31-99	25.00	33.93	18.75	25.00	50.00	28.12
	Female	Overall	60.04	62.88	66.38	66.04	69.17	64.24
	Female	10-30	60.32	63.16	66.71	66.07	69.18	64.66
	Female	31-99	58.50	56.89	59.90	66.67	68.32	56.71
	Overall		60.59	63.67	67.00	66.39	69.58	64.74

Table 19: F1-Scores for the ICLE corpora

			Trustpilot - Trustpilot_P	Trustpilot_P - Trustpilot
SVM	Male	Overall	60.97	69.55
	Male	10-30	60.23	65.89
	Male	31-55	61.36	70.85
	Male	56-99	58.05	74.63
	Female	Overall	62.10	66.50
	Female	10-30	56.82	63.93
	Female	31-55	69.41	71.24
	Female	56-99	60.29	57.78
	Overall		61.24	68.84
BERT	Male	Overall	66.20	71.90
	Male	10-30	61.24	61.66
	Male	31-55	68.04	77.47
	Male	56-99	75.56	75.72
	Female	Overall	63.76	71.73
	Female	10-30	65.47	66.81
	Female	31-55	70.09	76.81
	Female	56-99	62.12	71.17
	Overall		65.65	71.92

Table 20: F1-Scores for the Trustpilot corpora

		Paraphrased	Length	Gender	BG	CN	CZ	DE	DU	ES	FI	FR	IT	JP	NO	PL	RU	SE	TR	R^2	Acc.
ICLE - ICLE_P																					
SVM	Coef.	0.89***	0.00**	0.2	-0.94***	1.58***	0.07	0.01	-0.33	-0.3	-0.65*	-0.73***	0.43	1.24***	-1.01***	-1.04***	0.27	-0.45**	-0.16	0.15	0.78
	DA	0.03	0.01	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.02	0.02	0.00	0.00	0.00		
BERT	Coef.	0.49***	0.00**	0.17	0.02	2.63***	-0.37	-0.28	-1.25***	-0.21	-1.68***	-1.08***	1.08***	0.98***	-1.05***	-1.55***	0.37	-1.4***	0.79*	0.25	0.75
	DA	0.01	0.01	0.00	0.00	0.04	0.00	0.00	0.01	0.00	0.03	0.01	0.00	0.00	0.01	0.05	0.00	0.04	0.00		
ICLE - ICLE_S																					
SVM	Coef.	0.52***	0.00	0.39**	-1.03***	1.64***	-0.08	-0.11	-0.49*	0.2	-0.74**	-0.83***	0.19	1.02**	-0.95***	-0.04	-0.18	-0.36	-0.35	0.11	0.8
	DA	0.01	0.00	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00		
BERT	Coef.	0.45***	0.00*	0.17	0.23	2.3***	-0.18	-0.06	-1.34***	-0.03	-1.57***	-0.96***	1.19***	0.38	-1.21***	-1.14***	0.11	-1.37***	1.0**	0.21	0.75
	DA	0.01	0.01	0.00	0.00	0.04	0.00	0.00	0.02	0.00	0.03	0.01	0.01	0.00	0.02	0.02	0.00	0.04	0.00		
ICLE_P - ICLE																					
SVM	Coef.	-0.02	0.00	0.2	-0.89***	1.85***	-0.77***	-0.36*	-0.28	0.06	-0.32	-0.6**	0.08	1.1***	-0.61**	0.04	-0.14	-0.12	-0.39*	0.10	0.72
	DA	0.00	0.00	0.00	0.02	0.03	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00		
BERT	Coef.	0.14	0.00	0.27*	0.12	3.09***	-0.83***	-0.02	-1.34***	-0.93***	-0.67**	-1.11***	1.13***	1.1***	-0.94***	-0.03	-0.36	-0.61***	0.18	0.18	0.72
	DA	0.00	0.01	0.00	0.00	0.06	0.01	0.00	0.02	0.01	0.01	0.02	0.01	0.01	0.01	0.00	0.00	0.01	0.00		
ICLE_P - ICLE_S																					
SVM	Coef.	0.12	0.00	0.18	-0.83***	1.83***	-0.97***	-0.21	-0.33	0.13	-0.34	-0.45*	0.54*	0.93***	-0.77***	0.03	-0.24	-0.21	-0.48**	0.10	0.72
	DA	0.00	0.00	0.00	0.02	0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00		
BERT	Coef.	-0.06	0.00	0.27*	-0.01	3.0***	-0.78**	0.12	-0.95***	-0.8***	-0.8***	-0.82***	1.12***	0.89***	-1.01***	0.08	-0.34	-0.68***	-0.16	0.16	0.71
	DA	0.00	0.01	0.00	0.00	0.06	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.00	0.01	0.00		
ICLE_S - ICLE																					
SVM	Coef.	0.25*	0.00	0.39**	-0.95***	1.52***	-0.65**	-0.12	-0.29	-0.33	-0.55*	-0.7***	-0.18	1.53***	-0.37	0.25	-0.01	-0.13	-0.44*	0.09	0.77
	DA	0.00	0.00	0.00	0.02	0.02	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00		
BERT	Coef.	0.05	0.00	0.31*	-0.06	3.3***	-0.18	-0.48**	-1.37***	-1.34***	-0.89***	-1.27***	0.87***	1.22***	-0.55**	-0.36	-0.35	-0.76***	0.59*	0.18	0.74
	DA	0.00	0.01	0.00	0.00	0.06	0.00	0.01	0.02	0.02	0.01	0.03	0.00	0.01	0.00	0.00	0.00	0.01	0.00		
ICLE_S - ICLE_P																					
SVM	Coef.	-0.25*	0.00	0.23	-0.85***	1.77***	-0.4	0.18	-0.83***	-0.86***	-0.43	-0.67***	0.26	1.55***	-0.41*	-0.81***	0.3	0.13	-0.48**	0.12	0.72
	DA	0.00	0.01	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.00		
BERT	Coef.	-0.21*	0.00	0.24	-0.15	3.37***	-0.25	-0.4**	-0.97***	-1.0***	-1.0***	-0.83***	0.65**	1.19***	-0.82***	-0.86***	-0.43*	-0.91***	0.77**	0.18	0.71
	DA	0.00	0.01	0.00	0.00	0.06	0.00	0.00	0.01	0.01	0.01	0.01	0.00	0.01	0.01	0.01	0.00	0.02	0.00		

Table 21: Results from regression model and DA for ICLE corpora

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Druck-Exemplaren überein.

Datum und Unterschrift:

Declaration

I hereby declare that the work presented in this thesis is entirely my own. I did not use any other sources and references that the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted hard copies.

Date and Signature:

References

Sam Altman. Twitter, 2022. URL https://twitter.com/sama/status/1599668808285028353.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *CoRR*, abs/2004.05150, 2020. URL https://arxiv.org/abs/2004.05150.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. Toefl11: A corpus of non-native english. *ETS Research Report Series*, 2013(2):i–15, 2013.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna M. Wallach. Language (technology) is power: A critical survey of "bias" in NLP. *CoRR*, abs/2005.14050, 2020. URL https://arxiv.org/abs/2005.14050.

Julian Brooke and Graeme Hirst. Robust, lexicalized native language identification. In *Proceedings of COLING 2012*, pages 391–408, 2012.

Julian Brooke and Graeme Hirst. Native language detection with 'cheap'learner corpora. In *Twenty Years of Learner Corpus Research. Looking Back, Moving Ahead: Proceedings of the First Learner Corpus Research Conference (LCR 2011)*, volume 1, page 37. Presses universitaires de Louvain, 2013.

David V Budescu. Dominance analysis: a new approach to the problem of relative importance of predictors in multiple regression. *Psychological bulletin*, 114(3):542, 1993.

X Chen, P Ender, M Mitchell, and C Wells. Additional coding systems for categorical variables in regression analysis. *Regression with SPSS (chap. 5)*, 2011.

Marie Herget Christensen, Tanya Karoli Christensen, and Torben Juel Jensen. Foregrounding of subordinate clauses by word order: Psycholinguistic evidence of the function of v¿adv (v2) word order in danish. *Linguistics*, 58(1):245–273, 2020. doi: doi:10.1515/ling-2019-0040. URL `https://doi.org/10.1515/ling-2019-0040`.

Andrea Cimino and Felice Dell'Orletta. Stacked sentence-document classifier approach for improving native language identification. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 430–437, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5049. URL `https://aclanthology.org/W17-5049`.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

Council of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division. *Common European framework of reference for languages: Learning, teaching, assessment*. Cambridge University Press, 2001.

Erenay Dayanik, Ngoc Thang Vu, and Sebastian Padó. Analysis of bias in nlp models with regression and effect sizes. *Northern European Journal of Language Technology*, 8(1), 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423`.

Jacob Eisenstein. *Introduction to natural language processing*. MIT press, 2019.

Andreas Fuster, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther.

Predictably unequal? the effects of machine learning on credit markets. *The Journal of Finance*, 77(1):5–47, 2022.

Binyam Gebrekidan Gebre, Marcos Zampieri, Peter Wittenburg, and Tom Heskes. Improving native language identification with TF-IDF weighting. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 216–223, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL `https://aclanthology.org/W13-1728`.

Jeroen Geertzen, Theodora Alexopoulou, Anna Korhonen, et al. Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*, pages 240–254. Citeseer, 2013.

Google. Bert, 2023. URL `https://github.com/google-research/bert`.

Sylviane Granger, Estelle Dagneaux, Fanny Meunier, Magali Paquot, et al. *International corpus of learner English*, volume 2. Presses universitaires de Louvain Louvain-la-Neuve, 2009.

Frank E Harrell et al. *Regression modeling strategies: with applications to linear models, logistic regression, and survival analysis*, volume 608. Springer, 2001.

Zdzisław Hellwig. *Linear Regression and its application to economics*. Elsevier, 2014.

Dirk Hovy. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 752–762, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1073. URL `https://aclanthology.org/P15-1073`.

Dirk Hovy, Anders Johannsen, and Anders Søgaard. User review sites as a resource for large-scale sociolinguistic studies. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, page 452–461, Republic

and Canton of Geneva, CHE, 2015. International World Wide Web Conferences Steering Committee. ISBN 9781450334693. doi: 10.1145/2736277.2741141. URL https://doi.org/10.1145/2736277.2741141.

Shin'ichiro Ishikawa. A new horizon in learner corpus studies: The aim of the icnale project. *Korea*, 404:89–168, 2011.

Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. Gpt-4 passes the bar exam. *Available at SSRN 4389233*, 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9), jan 2023. ISSN 0360-0300. doi: 10.1145/3560815. URL https://doi.org/10.1145/3560815.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. Gender bias in neural natural language processing. *Logic, Language, and Security: Essays Dedicated to Andre Scedrov on the Occasion of His 65th Birthday*, pages 189–202, 2020.

Shervin Malmasi and Mark Dras. Language transfer hypotheses with linear svm weights. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1385–1390, 2014.

Shervin Malmasi and Mark Dras. Native language identification with classifier stacking and ensembles. *Computational Linguistics*, 44(3):403–446, 09 2018. ISSN 0891-2017. doi: 10.1162/coli_a_00323. URL https://doi.org/10.1162/coli_a_00323.

Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. A report on the 2017 native language identification shared task. In *Proceedings of the 12th Workshop*

on Innovative Use of NLP for Building Educational Applications, pages 62–75, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5007. URL https://aclanthology.org/W17-5007.

Christopher Manning and Hinrich Schutze. *Foundations of statistical natural language processing*. MIT press, 1999.

Daniel McFadden et al. Conditional logit analysis of qualitative choice behavior. 1973.

Josh Meyer, Lindy Rauchenstein, Joshua D. Eisenberg, and Nicholas Howell. Artie bias corpus: An open dataset for detecting demographic bias in speech applications. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6462–6468, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.796.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL https://arxiv.org/abs/1301.3781.

Mirza Alim Mutasodirin and Radityo Eko Prasojo. Investigating text shortening strategy in bert: Truncation vs summarization. In *2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 1–5. IEEE, 2021.

OpenAI. Customer stories, 2023. URL https://openai.com/customer-stories.

Ria Perkins. *Linguistic identifiers of L1 Persian speakers writing in English: NLID for authorship analysis*. PhD thesis, Aston University, 2014.

Ella Rabinovich, Yulia Tsvetkov, and Shuly Wintner. Native Language Cognate Effects on Second Language Lexical Choice. *Transactions of the Association for Computational Linguistics*, 6:329–342, 05 2018. ISSN 2307-387X. doi: 10.1162/tacl_a_00024. URL https://doi.org/10.1162/tacl_a_00024.

Gopinath Rebala, Ajay Ravi, Sanjay Churiwala, Gopinath Rebala, Ajay Ravi, and Sanjay Churiwala. Machine learning definition and basics. *An introduction to machine learning*, pages 1–17, 2019.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

Sabrina Stehwien and Sebastian Padó. Generalization in native language identification: Learners versus scientists. *CLiC it*, page 264, 2015.

Sabrina Stehwien and Sebastian Padó. Native language identification across text types: How special are scientists? *IJCoL. Italian Journal of Computational Linguistics*, 2(2-1), 2016.

Stian Steinbakken and Björn Gambäck. Native-language identification with attention. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 261–271, Indian Institute of Technology Patna, Patna, India, December 2020. NLP Association of India (NLPAI). URL `https://aclanthology.org/2020.icon-main.35`.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune BERT for text classification? *CoRR*, abs/1905.05583, 2019a. URL `http://arxiv.org/abs/1905.05583`.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth M. Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. *CoRR*, abs/1906.08976, 2019b. URL `http://arxiv.org/abs/1906.08976`.

Joel Tetreault, Daniel Blanchard, Aoife Cahill, and Martin Chodorow. Native tongues, lost and found: Resources and empirical evaluations in native language identification. In *Proceedings of COLING 2012*, pages 2585–2602, 2012.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. A report on the first native language identification shared task. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications*, pages 48–57, 2013.

Christopher Glen Thompson, Rae Seon Kim, Ariel M Aloe, and Betsy Jane Becker. Extracting the variance inflation factor and other multicollinearity diagnostics from typical regression results. *Basic and Applied Social Psychology*, 39(2):81–90, 2017.

H. Holden Thorp. Chatgpt is fun, but not an author. *Science*, 379(6630):313–313, 2023. doi: 10.1126/science.adg7879. URL https://www.science.org/doi/abs/10.1126/science.adg7879.

Enrica Troiano, Laura Oberländer, and Roman Klinger. Dimensional Modeling of Emotions in Text with Appraisal Theories: Corpus Creation, Annotation Reliability, and Prediction. *Computational Linguistics*, 49(1):1–72, 03 2023. ISSN 0891-2017. doi: 10.1162/coli_a_00461. URL https://doi.org/10.1162/coli_a_00461.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: The impact of student initialization on knowledge distillation. *CoRR*, abs/1908.08962, 2019. URL http://arxiv.org/abs/1908.08962.

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, 1974. doi: 10.1126/science.185.4157.1124. URL https://www.science.org/doi/abs/10.1126/science.185.4157.1124.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Randy Joy Magno Ventayen and Caren C Orlanda-Ventayen. Graduate students' perspective on the usability of grammarly® in one asean state university. *Asian ESP Journal*, 14(7.2), 2018.

Pranshu Verma and Will Oremus. Chatgpt invented a sexual harassment scandal and named a real law prof as the accused. *The Washington Post*, 2023. URL `https://www.washingtonpost.com/technology/2023/04/05/chatgpt-lies/`.

Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93, 2016.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.