Bachelor's Thesis

Investigating the Influence of Machine Translation on Sentiment: A Machine Learning Approach for Multilingual Sentiment Analysis

Author
Sarah Ackerschewski
sarah.ackerschewski@student.uni-tuebingen.de

Supervisor
Dr. Çağri Çöltekin
ccoltekin@sfs.uni-tuebingen.de

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

Bachelor of Arts
in
International Studies in Computational Linguistics

Seminar für Sprachwissenschaft
Eberhard Karls Universität Tübingen

March 2022

Antiplagiatserklärung

Ich erkläre hiermit,

dass ich die vorliegende Arbeit selbständig verfasst habe,

dass ich keine anderen als die angegebenen Hilfsmittel und Quellen benutzt habe,

dass ich alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet habe,

dass die Arbeit weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen ist,

dass ich die Arbeit weder vollständig noch in wesentlichen Teilen bereits veröffentlicht habe,

dass das in Dateiform eingereichte Exemplar mit dem eingereichten gebundenen Exemplar übereinstimmt.

Tübingen, den 3. März 2022

Sarah Ackerschewski

Contents

List of Tables

List of Figures

Abstract

The purpose of this study is to investigate the effects of translation loss on sentiment polarity classification. German movie reviews were translated to English with the DeepL API, i.e. with a machine translation system. The data was used as test data for a sentiment analyzer that is trained on English movie reviews. For comparison, English test data and a different machine translator on the German movie reviews were also applied. The results were evaluated with precision, recall, F1-measure, and accuracy. The translations and the predictions were manually annotated in regards to mistranslations and misclassifications.

The results provide support for previous research outcomes that machine-translated data can be employed as test data for sentiment analysis and performs similar to English test data, as all evaluation metrics only drop about 3% and only 3.26% of all misclassifications may be caused by incorrect translations. In addition, the comparison with the other translations revealed that the quality of the translation is important for sentiment analysis, thus, proving that translation loss impacts the sentiment classification, nonetheless. The thesis concludes with further research questions that need to be examined in the future.

1 Introduction

Sentiment Analysis, or Opinion Mining, is a common technique in the field of Natural Language Processing (NLP). It focuses on identifying the sentiment on a certain topic based on a text (Medhat et al., 2014). Especially because of the increase of using social media, blogs, forums, and other platforms for expressing one's opinion, more textual data is provided for NLP research, thus also for text mining investigations such as sentiment analysis.

Therefore, sentiment analysis, in particular subjectivity and polarity classification, is already well studied. However, these studies mostly focus on English data as it offers a lot of resources to work with (Can et al., 2018). Other languages that do not provide much data to use for NLP tasks or in particular sentiment analysis are less researched. Various techniques are already proposed by different research groups for different languages. There are approaches using a machine translation system to translate the low-resource language into a language with a lot of textual data, i.e. English, and then using it as test data for a machine learning model, trained to classify given sentences as positive or negative with English training data (Can et al., 2018; Mohammad et al., 2016). Others tried out unsupervised learning with neural network techniques that try to recognize the patterns of the data without having correct labels in the input (Feldman, 2013). There are also approaches, which combine both techniques or enhance each technique with sentiment lexica (Feldman, 2013). Nonetheless, it seems that there is no clear preferred method regarding multilingual sentiment analysis. Thus, it is still quite an important research topic in NLP.

Sentiment analysis is especially popular for finding out how the opinion of others can influence our own decisions regarding that product, news, person, etc. (Feldman, 2013). It can be assumed that when we see many bad movie reviews, it is not very likely that we go and see that movie. Vice versa, we will probably be more likely to watch a movie, when hearing much praise.

Companies also gain insight into what people think about their products or company in general. With sentiment analysis, they are able to manage these better than assigning a person with the task of "monitoring" the internet (Feldman, 2013).

This paper aims to find out how a machine learning model performs with machine-translated test data from a low-resource language compared to original English data on the basis of movie reviews, which seem quite popular as data for opinion mining (e.g. used by Hutto and Gilbert (2014)). As mentioned, a few studies (Can et al., 2018; Mohammad et al., 2016) already explored such approaches. However, the low-resource languages chosen for these experiments often were Arabic, Spanish, Turkish, Dutch, or Russian that were translated to English. German data that is translated into English seems to be conducted less, even though the German data for sentiment analysis is also rare. Therefore, this study will explore how the English machine translation of German data influences the polarity predictions of a machine learning sentiment analyzer and if the results of a machine translation system, which translates from German to English, are competitive. If such is the case, there would be a new possibility of providing data for the - in this field - low-resource language German. Based on other experiments that examined the same question but for different languages, it may be presumed that these findings hold for the German language as well.

Furthermore, Vanmassenhove et al. (2019) learned that the translation with a machine translation system will tend to generalize its translations, such that lexical diversity decreases with the translation. On the basis of this observation, this paper will also investigate in this direction to see if especially this "type" of translation loss, i.e. the quality of the translation, impacts the sentiment polarity classification.

Since the research question here is to find out if machine translation and therefore the consequential translation loss impacts the sentiment classification process, two different machine translation systems will be used. This way one can hypothesize that the "worse" translation will also perform worse in the sentiment analyzer, while the other translation will yield better polarity predictions. Such results would demonstrate that the machine translation, or more precisely the translation loss caused by the lower quality, has an impact on the sentiment analysis.

First, this paper will review various techniques of previous research in section 2. Then, it goes on to the overall method used in order to investigate these hypotheses. The method section will also cover the data which is used and the applied pre-processing of this. The sentiment analyzer, which is built based on the various approaches illustrated in section 2 and used for classifying sentence-level snippets into "positive" and "negative", is explained in section 3.3. The evaluation of this model and its results are also presented. Finally, the results will be discussed to see whether the hypothesis was verified or falsified.

2 Background

Sentiment analysis comprises different classification tasks for identifying the overall sentiment of either a document (document-level sentiment analysis), a sentence (sentence-level sentiment analysis), or aspects of a product (aspect-based sentiment analysis) (Feldman, 2013).

Usually, the first or most common classification task is to distinguish whether the input is objective or subjective (Feldman, 2013). If the classification result is subjective, a further classification between negative and positive can be made or a model yields a so-called sentiment score instead (Feldman, 2013). This sentiment score can be defined as "a measurement of human opinions and affective states" (Tian et al., 2018, p.1) and is usually represented with a scale from strongly negative sentiment to strongly positive sentiment (Tian et al., 2018). For example, Hutto and Gilbert (2014) created a sentiment valence/intensity dictionary called VADER and thus, made use of a sentiment score in the form of a scale from -4 (strongly negative) to +4 (strongly positive).

A common sentiment classification "template" that describes the procedure is given by Feldman (2013). He explains that at first, the input, which usually is a corpus of documents, is read into the program and the resulting data is pre-processed. Conventional approaches include stemming or lemmatization, word or sentence tokenization, part-of-speech (POS) tagging, and various more (Feldman, 2013). The main part of the sentiment classification "template" is the sentiment analyzer, which is a machine learning approach most of the time. There are three different machine learning approaches commonly applied in sentiment analysis. One is supervised machine learning, where the model is trained on pre-classified data and is thus able to make classifications on unseen data. When no such pre-classified data, also called training data, is available the average semantic orientation of specific phrases within a document is calculated (Feldman, 2013). This unsupervised machine learning approach often utilizes pointwise mutual information in order to calculate the semantic orientation (Feldman, 2013). The last possibility is a combination of both approaches. Semi-supervised machine learning takes labeled as well as unlabeled data and sometimes combines lexica with machine learning (Ahmad and Aftab, 2017). These three machine learning approaches are able to yield the sentiment polarity, i.e. they can classify whether the input is negative or positive (and in some cases neutral). These classification results could not only be helpful for companies seeking feedback about their products or image but also for everyday people seeking advice from other internet users regarding certain products, persons, or news (Feldman, 2013).

As previously noted, sentiment analysis for polarity classification on English textual data is already well-studied with various techniques applied. Medhat et al. (2014), who reviewed the sentiment analysis research field so far, claim that support vector machines (SVM) is one of the most frequently used techniques for sentiment polarity detection.

Ahmad and Aftab (2017) tested the performance of SVM's in various conditions, such as binary and multiclass, i.e. negative, positive, and neutral classification with three different datasets and varying training to test data ratios. 70% training data to 30% test data, 50% to 50%, and 30% to 70% were the possible ratios for splitting the training and test data. The first data set contained Tweets about "Apple", "Google", "Microsoft", and "Twitter", the second one contained Tweets about all major U.S. airlines and the third contained movie reviews from IMDb. Evaluation of the SVMs with the three datasets and the three different ratios showed that the SVM performance depends not only on the dataset it is trained and tested on but also on the ratio between the size of the training data and the size of the test data (Ahmad and Aftab, 2017). This was apparent when looking at the average Precision and Recall of the first data set, which yielded the best results with a train/test data split of 70:30, compared to the Precision and Recall of the third data set, which had its best results when the data was split 50:50 (Ahmad and Aftab, 2017).

Ahmad and Aftab (2017) also mention the importance of pre-processing, which is even further exemplified in Haddi et al. (2013). They also chose an SVM sentiment analyzer and were able to achieve an accuracy of 92.3% after applying various pre-processing approaches, such as the expansion of abbreviations, stop word removal, stemming, and creating a feature matrix with TF-IDF (Haddi et al., 2013). However, Haddi et al. (2013) draw attention to feature selection as well. In their sentiment framework, univariate chi-squared was applied and the feature matrix was filtered with a significance level of 95%. In comparison, the 92.3% accuracy they achieved with pre-processing and feature selection, would be at 81.5% without the feature selection and at 78.33% if the data would not have been pre-processed at all, but only transformed into a feature matrix (Haddi et al., 2013). Thus, their research demonstrates the positive effect of pre-processing and feature selection on sentiment analysis.

Another previous study that is, however, closer to the research question raised in this paper was done by Balahur and Turchi (2012). In this study, the focus lies on different Machine Translation systems and if their results are suitable for sentiment analysis, which was again examined by using an SVM. They used the translations from English to French, German, and Spanish to produce training and test data in these languages. The application of the German, Spanish and French training and test data resulted in a weighted F1-measure of 0.685, 0.636, and 0.652, respectively. The results demonstrated that a poor translation could cause a bad performance as it increases the variance and more sparse feature vectors (Balahur and Turchi, 2012). Still, the authors consider Machine Translation an option in order to create training data for low-resource languages (Balahur and Turchi, 2012).

As Balahur and Turchi (2012) observed, the quality of the translation has an impact on the sentiment analysis. Thus looking at Machine Translation and the causes for that is also of importance. Translation Loss in Machine Translation was investigated more broadly by Vanmassenhove et al. (2019). They claim that the often extreme loss of lexical richness and diversity indicated that there is an "underlying" problem with the translation system. Vanmassenhove et al. (2019) compared two different Machine Translation techniques, Neural Machine Translation, and Statistical Machine Translation, with human translation, and came to the conclusion that Statistical Machine Translation performed better than Neural Machine Translation although the human translation was still slightly better. Nonetheless, Statistical Machine Translation approaches only try to reach the highest probability for a sentence, less frequent words are disregarded (Vanmassenhove et al., 2019). These disregarded words, however, could increase the lexical diversity, and this is quite important for data mining tasks (Vanmassenhove et al., 2019).

The translation loss and especially its influence on sentiment was also researched by Mohammad et al. (2016). According to them, translating either the focus-language data to English and using it on a machine learning model trained with English data or translating English resources such as lexicons to the focus-language and using it to build a machine learning model with the focus-language are the two most widely used methods for applying sentiment analysis on low-resource languages. They examined both procedures separately and found that the first procedure, translating the low-resource language to English and using it as test data for an English sentiment analyzer, is reasonable compensation for using original data and an original sentiment analyzer. A reason for that could be the "consistent" mistakes that are made by a machine translation system (Mohammad et al., 2016). This way the sentiment analyzer can still use the wrong translation as a prompt to detect the sentiment (Mohammad et al., 2016). At the same time, the use of translated resources, i.e. from English to the low-resource language with a sentiment analyzer trained on the low-resource language produced a lower accuracy (Mohammad et al., 2016).

In a similar manner, a multilingual deep learning approach with a neural network sentiment analyzer is explored by Can et al. (2018). They chose a recurrent neural network (RNN) using text and pre-trained word embeddings with an embedding length of 100 as input. The training data consists of higher weighted restaurant reviews as well as lower weighted reviews from other domains. After applying the test data consisting of Spanish, Turkish, Dutch, and Russian restaurant reviews, which were translated to English with the Google Translate API, Can et al. (2018) were able to achieve up to 85% accuracy.

There are also a few attempts that deal with German data, which are, however, limited to lexicon-based approaches. For instance, Waltinger (2010) created a sentiment polarity lexicon, GermanPolarityClues, with the focal point on "polarity features", i.e. words that carry positive or negative sentiment. For this, two dictionaries that were already translated to German in a previous study by Waltinger (2010) were extended with common negation phrases and synonyms of positive and negative features, which were not included so far (Waltinger, 2010). The translated features got the sentiment "label" from the original English dictionary. With the intention of capturing ambiguities, such features were handled manually (Waltinger, 2010). The GermanPolarityClues dictionary was able to reach an F1-measure of about 88% with data from Amazon Product reviews when applied to a Linear SVM (Waltinger, 2010).

Other work, which was found during the research for this paper, also rather focussed on sentiment lexica than on approaches mentioned above for polarity classification, such as Tymann et al. (2019) creating a German equivalent to the VADER dictionary by Hutto and Gilbert (2014) or Schmidt et al. (2021) developing a lexicon-based sentiment analyzer for finding the intensity of a sentence.

3 Method

The sentiment analyzer should work as shown in Figure 1. A Support Vector Machine (SVM) is trained on pre-processed feature vectors of sentence-level snippets of movie reviews, which are already collected and pre-classified into "positive" and "negative" by Hutto and Gilbert (2014). The German movie reviews are tokenized on a sentence level, where each sentence level snippet is given the general sentiment of the review for evaluation later on, in order to create the test data. The gold standard is not ideally created this way, but unfortunately, no other German movie review dataset, already classified on a sentence level, was found. Afterward, the German data is translated to English via DeepL as well as with a pre-trained machine translation model. Thereafter, all pre-processing steps stated in 3.2. are applied to the English training data and the translated test data. The created feature vector can be used with the sentiment analyzer for testing, thus being classified into "positive" and "negative".

With the intention of seeing if there is an impact of the translation loss from German to English, English movie reviews are tokenized, assigned the review label, and pre-processed, like the German reviews as well, to create the English test data (see 3.1). Consequently, the corresponding feature vector is also used as test data for the same classifier, and thus, the English movie review snippets are classified into the two polarity classes, too.

All the details of the sentiment analyzing framework, such as the data used, the pre-processing techniques applied, and the tuning of the SVM, are described in the following subsections. Also, the complete source code is made available on Github [1].

[1] https://github.com/sarahackerschewski/sentiment-translation-loss

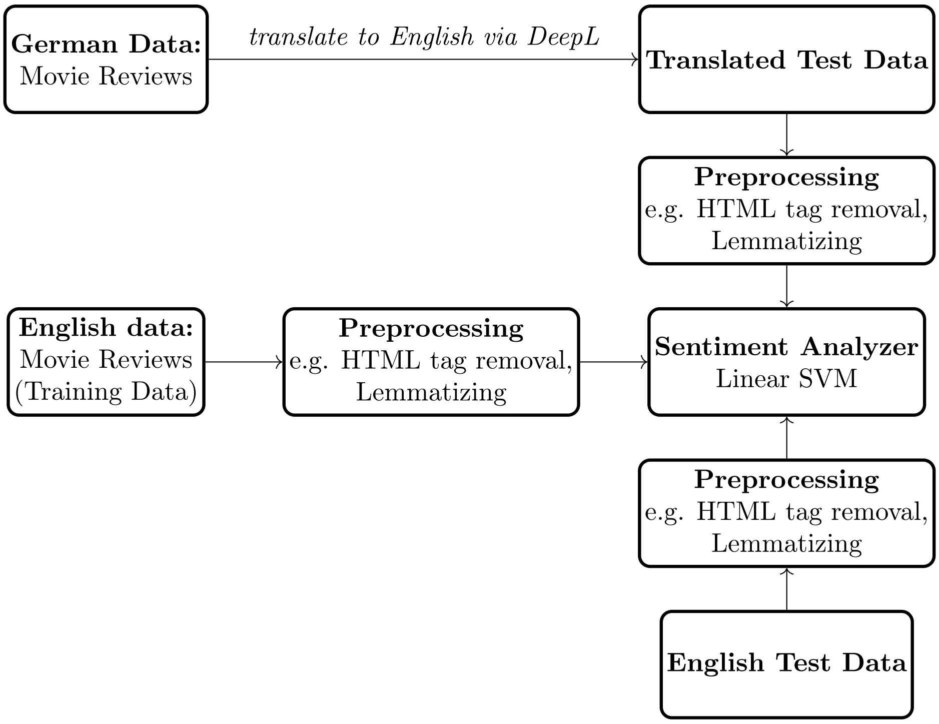

Figure 1: Flowchart of the Framework

3.1 Data

Since it is a common domain and German test data, as well as English training and test data, were found, movie reviews were chosen as the domain for training and testing the sentiment analyzer. The model was trained on pre-classified data from Hutto and Gilbert (2014). They tokenized 2000 reviews (Pang and Lee (2004), as cited in Hutto and Gilbert 2014) yielding 10.605 sentence-level snippets that were rated by 20 independent pre-screened human annotators on a scale from -4 (extremely negative) to +4 (extremely positive). For training purposes, all snippets that were rated from -4 to -1 got the new label "negative", while +1 to +4 rated snippets got the new label "positive". As the focus of this experiment lies on binary classification, a neutral label for the snippets rated 0 was not given.

In order to test the sentiment analyzer, English test data was taken from Maas et al. (2011) in the form of the "Largest Movie Review Database", which contains movie reviews from the website IMDb, which is an "online database of information related to films, television series, home videos, video games, and streaming content online"[2]. Each review was tokenized, randomized, and assigned the overall sentiment of the review for evaluation purposes. The final English test data comprised 1000 positive and 1000 negative sentences. The German test data was gathered from Guhr et al. (2020), who studied German sentiment analysis with different domains, amongst them movie reviews. The "Filmstarts" reviews, which is a German website similar to IMDb, were also tokenized, randomized, and given the overall sentiment from the pre-classified review. The final German test data also contained 1000 positive and 1000 negative sentence-level snippets.

The set of German sentences was then translated to English with the DeepL API[3] in order to utilize it with the sentiment analyzer. The same set of sentences is also translated with a modified approach of a "Towards Data Science" article[4]. In short, a Pytorch transformer pipeline is making use of a pre-trained machine translation model found on HuggingFace [5] in order to translate the German test data to English, thereby creating a different translation for

[2] https://en.wikipedia.org/wiki/IMDb

[3] https://www.deepl.com/translator

[4] https://towardsdatascience.com/machine-translation-with-transformers-using-pytorch-f121fe0ad97b

[5] https://huggingface.co/Helsinki-NLP/opus-mt-de-en

comparative purposes. That way, the predictions for the DeepL translations can then be seen in contrast to the predictions for the other translations.
The sentiment analyzer's performance was also examined with English test data from another domain as a side exploration. The test data for this purpose were English hotel reviews that were taken from the four-city dataset [6]. Then, each review was tokenized, assigned gold labels, and randomized as done for the other test sets, too. The resulting test data was also checked for the language the review was written in because the hotel reviews were sometimes in French, German, or another language. For testing, again, 1000 positive and 1000 negative sentences were used.
For tokenizing all test sets, the NLTK tokenizer (Bird et al., 2009) was applied. This is the same tokenizer that Hutto and Gilbert (2014) used for their sentence-level snippets.

3.2 Pre-Processing

As stated in all previous studies mentioned above, pre-processing is an important step before building a machine learning model. The pre-processing can be split into data transformation and feature selection because feature selection is not always applied. Considering that the data often still contains HTML tags, these have to be removed to transform the data and reduce the noise and sparseness of the feature vector built with the data. Another step taken is the lemmatization of each word. Lemmatization is a technique for reducing each word to its base form, e.g. "loved" is being reduced to "love". Here the WordNetLemmatizer[7] from NLTK was applied. Such an approach helps to decrease the redundancy of words and thus decrease the number of features (Haddi et al., 2013).
Further pre-processing steps were not taken because the textual data contains a lot of clues for correctly classifying the test data into positive or negative. The removal of stopwords, for example, is not beneficial if you take into account that words such as "not", "no" or "but" are regarded as stopwords. These words do not necessarily carry sentiment, but they can reverse the sentiment given by other words. Some studies also removed non-alphabetic characters from the sentences, e.g. Haddi et al. (2013). Nevertheless, this was not applied in this study as non-alphabetic characters often convey sentiment as well and other emoticons, such as ":)" and ":(" which convey "positive" polarity and "negative" polarity, respectively. The model could pick up these combinations to make better classifications. Bahrainian and Dengel (2013) also achieved the best results when keeping the stopwords and emoticons. Furthermore, in movie reviews, numbers can be helpful clues as well, because movies are often associated with a number as their rank. For instance, "I give this movie 5 out of 5 stars" and "I give this movie 1 out of 5 stars" could not be distinguished if numbers are removed as a pre-processing step.
The final data transformation step is creating a feature vector with the Term Frequency-Inverse Document Frequency (TF-IDF) of sci-kit-learn (Pedregosa et al., 2011). For creating the TF-IDF vectors, first, the frequency of each term in the given document is calculated (Term-Frequency) and then the frequency is reversed to weigh rare terms higher than very common ones (Inverse-Document-Frequency) (Ahmad and Aftab, 2017). The feature vector consists of uni-, bi-, and trigrams. An n-gram is a simple combination of n words in the order they appeared (Jurafsky and Martin, 2021). In a sentence such as "I did not like this movie", a unigram splits the sentences into each term ("I", "did", "not", "like", "this", "movie"), a bigram separates two words at once ("I did", "did not", "not like", "like this", "this movie") and a trigram separates three words at once ("I did not", "did not like", "not like this", "like this movie"). Thus, using uni- to trigrams may capture longer dependencies between the word and phrases and in the example given above is able to recognize the negative sentiment that "not like" conveys (Balahur and Turchi, 2012). Nevertheless, Balahur and Turchi (2012) recommend not using higher-level n-grams since it increases the sparseness of the feature vectors.
As mentioned by Haddi et al. (2013), feature selection can help improve the classification since the model does not have to work with too many features. However, it can be argued that all features are important for the model to memorize the correct clues. In this experiment, the

[6] https://www.cs.cmu.edu/~jiweil/html/hotel-review.html
[7] https://www.nltk.org/api/nltk.stem.wordnet.html

best 10-fold CV showed that the feature selection was beneficial as it increases the 10-fold CV accuracy from 79% to 83%. Following the approach by Haddi et al. (2013), a chi-squared feature matrix was created, i.e. an array consisting of the chi-squared statistics of each feature. The features-to-keep were then chosen by the alpha, the "critical value" Haddi et al. (2013). Even though the best cross-validation accuracy results were achieved with quite low alphas (85% resulted in an 84% CV accuracy), the accuracy, when the test data was applied, decreased. This is why in this study it was decided to go no lower than 90% for the significance-level as the model presumably needs more features to make better classifications.

3.3 Sentiment Analyzer

Since it proved to be a reliable model for sentiment classification before (Bahrainian and Dengel (2013); Haddi et al. (2013); Mohammad et al. (2016); Waltinger (2010)), a linear Support-Vector Machine (SVM) was chosen for this research question. An SVM tries to establish the decision boundary between two classes, in this case, "positive" and "negative", in order to split the training data accordingly into the classes (Hastie et al., 2009). The decision boundary is also called a hyperplane and can be calculated with the following function (Hastie et al., 2009, p.418):

$$\{x : f(x) = x^T\beta + \beta_0 = 0\}$$

For every x in the "training data consist[ing] of N pairs $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$, with $x_i \in IR^p$ and $y_i \in \{-1, 1\}$", $f(x)$ is estimated as defined above and represents the "distance "(Hastie et al., 2009, p.418). For this polarity classification task it should be the case that $y_i \in$ negative, positive. Since the training data for this experiment is linearly separable, a linear SVM should be sufficient because it also permits minimalized overlaps for the data (Hastie et al., 2009).

Then, a 10-fold Cross-Validation (CV), which is a common measure to evaluate the model also used by other studies, e.g. Haddi et al. (2013); Mohammad et al. (2016), was applied for calculating the prediction error of the model (Hastie et al., 2009). More specifically, the 10-fold CV splits the training data into 10 parts using 9 parts for training and 1 part for testing, which is repeated until each part was used for validation once, so the average accuracy of all validation trials can be estimated (Hastie et al., 2009).

The SVM was also given a "tuning parameter" so that the test error is minimized (Hastie et al., 2009). According to Hastie et al. (2009), it is conventional for an SVM to set a higher "cost parameter" C. In this analyzer, the parameter C was "ideally" identified with a "manual grid search" at first, i.e. logarithmic Cs were chosen (from 0.0001 until 10000) and used for the analyzer. The C, which resulted in the best 10-fold CV accuracy, was explored a bit more to see if some "neighbor"-values were able to improve this accuracy. For the model to converge with higher C values as well, the maximum iteration had to be increased to 10000. Value 1000 performed the best with an accuracy of 82.82% and a standard deviation from this accuracy of 0.01. Although a C of 10000 resulted in the same accuracy and standard deviation, it was not selected because it took longer to converge. The "neighbor" values of 1000 also achieved the same accuracy and standard deviation, but because of the longer converging time, a lower C was chosen at first. The same procedure was followed a few times and after performing a sci-kit learn GridSearch with the last "neighbors", the value 375 was picked as the cost parameter C, which resulted in a CV accuracy of about 83% and a standard deviation of 0.01.

For comparison, the test data was also given to a sentiment lexicon to see, if the translation loss has or does not have an impact there too. In this case, the sentiment lexicon chosen was TextBlob (Loria, nd).

3.4 Evaluation Metrics

The data was used on the sentiment analyzer as described in Figure 1. The results of the experiment were evaluated with the common evaluation measures accuracy, precision, recall, and the F1-measure. The "ideal" labels for the test data were obtained from the data as well (see

section 3.1) and were referred to as the “gold” standard.
For calculating the measures, it is important to define the “true positives” (TP), which are the sentences that are correctly classified as positive by the sentiment analyzer and are also labeled positive in the gold standard (Jurafsky and Martin, 2021). This is the same for “true negatives” (TN), which are classified as negative and also have a gold label that is negative (Jurafsky and Martin, 2021). On the other hand, “false positives” (FP) are classified as positive but are actually negative, while “false negatives” (FN) are classified as negative but are actually positive (Jurafsky and Martin, 2021). Consequently, the precision estimates how many of the classified sentences are classified as positive by the model have “positive” gold labels (Jurafsky and Martin, 2021).

$$Precision = \frac{TP}{TP+FP}.$$

Recall then assesses how many of all classifications are positive in the gold standard (Jurafsky and Martin, 2021).

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

The harmonic mean of the precision and the recall gives the F1-measure (Jurafsky and Martin, 2021).

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

Finally, the accuracy simply captures how many classifications are the same in the gold standard. It should also be mentioned that all the following results are the macro-averaged evaluation outcomes, which is the unweighted mean for each metric of each class (Pedregosa et al., 2011).

3.5 Annotation

To establish whether the results of this experiment are a cause of translation loss, the test data and predictions were annotated by following the approach of Can et al. (2018). All mistranslations and misclassifications were marked and the proportion of all mistranslations to all misclassifications was calculated (Can et al., 2018).
While annotating, a mistranslation was either characterized as “not translated”, mistranslation (a), or “wrongly translated”, mistranslation (b). Not translated words are words still in the source language, German, and not in the target language, English. For instance, “seeehhhrr”, which literally translated means “veeerrryy”, or “spielts”, which should be changed to ”plays” by the machine translation system, were not translated to English but instead stayed the same in the sentence. It was noticeable that most words, which were not translated, were misspelled (“avnung” → “Ahnung” → “clue”/”idea”) or highly colloquial (“derbe” → “very”). On the other hand, words like “crass” from the German “krass” (“extreme”) do not exist in English, and wrong translations of German idioms such as “Erste Sahne” → “First cream” or “volle Lotte” → “full Lotte” often do not represent the sentiment of the original. Such examples were annotated as mistranslation (b) then. What is important to note here is that the wrongly translated words or grammatical mistakes were annotated by a non-native English speaker, thus they are purely subjective and may be annotated incorrectly.
In section 3.1 the unsuitability of the gold labels for evaluation is briefly mentioned. The gold standard of the English and German-to-English data is not necessarily correct since it is taken from the sentiment of the whole review. Hence, the wrong gold labels are assumed to cause a lot of the misclassifications of the data, which is why the test data, as well as the predictions, were later on annotated for the gold labels, too. That way it is attempted to identify how many “misclassifications” could actually be classified correctly by the sentiment analyzer. Nevertheless, the English and the German-to-English test data sets are created in a similar manner and therefore any comparison results of the predictions will hold.
While annotating the sentences based on the labels and the translations, other sentence proper-

ties were attracting some attention, too. Although some misclassifications should clearly be the opposite polarity, other sentences were too ambivalent out of context to choose if the predicted label is correct or incorrect, and yet others were clearly neutral or summaries of the movie, hence none of the present classes. The annotation of such sentences as having the wrong gold label is very subjective again, but might also show that the sentiment analyzer has similar problems if those are misclassified.

category	**example sentence**	**given gold label**
incorrect label	*witherspoon is wasted in this role.*	positive
	the end also very good!	negative
neutral sentence	*When the cats wake up, they start on a long trek home.*	positive
ambivalent sentence	*i rented this movie, but i wasn't too sure what to expect of it.*	positive

Table 1: Examples for possible misclassification causes

Table 1 holds examples for all categories the annotations of the predictions of both data sets, English, and German-to-English, were based on. In this table, I will not give any further examples for the mistranslations from German to English as some are already given above.

The example sentence in the first row is taken from the English test data and the overall positive sentiment of the review led to a positive gold label. The actual sentiment should be negative regardless. The most reasonable interpretation of this sentence could be that the movie is so bad that the talent of Witherspoon is "wasted". The word "waste" was subjectively annotated as a negative word but also verified as such by using the sentiment feature of TextBlob solely for the word "waste". The result of TextBlob is a -0.2 on a range from -1 to 1, thus indicating a negative polarity and a wrong gold label for this sentence.

The second example, taken from the German-to-English test data, for the "incorrect label" category shows the opposite. The sentence is assigned a negative gold label due to the whole negative review. The polarity of this sentence is given a 0.9 by TextBlob, which is almost the highest value possible. This can be explained by the word "good", which is a positive word. If working with sentiment intensity, the positive sentiment would even be increased by the adverb "very". Both examples might be classified correctly with their opposite gold label by the sentiment analyzer but are counted as misclassifications by the evaluation measures because their "official" gold label is the opposite polarity.

The neutral sentence, which is extracted from the English data, is most likely a snippet from the movie summary. Even if not, it seems that there is no clear polarity in this sentence. Instead, it is a subjective sentence. This is also confirmed by the TextBlob polarity score of -0.05, which is almost 0, i.e. neutral, and its subjectivity score of 0.4 on a range from 0 to 1 (in comparison, the subjectivity score of the second example is at 0.78). The positive gold label is not correct, but a correct classification could not be made by the sentiment analyzer either, which predicted the sentence's sentiment as negative. This is due to the negative-positive binary classification task of the SVM. Sentences like these could only be classified appropriately with a multi-class classification task that adds a neutral class.

Finally, the ambivalent category is purely subjective and based on the fact that some phrases could be the cue to a positive and a negative review. The given example from the English test data was taken from a positive review. However, on its own, it could have a positive but also a negative meaning. Without more context, the true polarity is subjective to the reader and thus can cause the sentiment analyzer trouble to make the "correct" prediction.

The annotation results and what they exactly mean will mainly be discussed in section 5.

4 Results

The research question of this thesis is based on translation loss in sentiment classification. Therefore, it follows to compare the evaluation of the original English data with the evaluation of the German data that was translated to English (shortened to German-to-English data in the rest of the thesis).

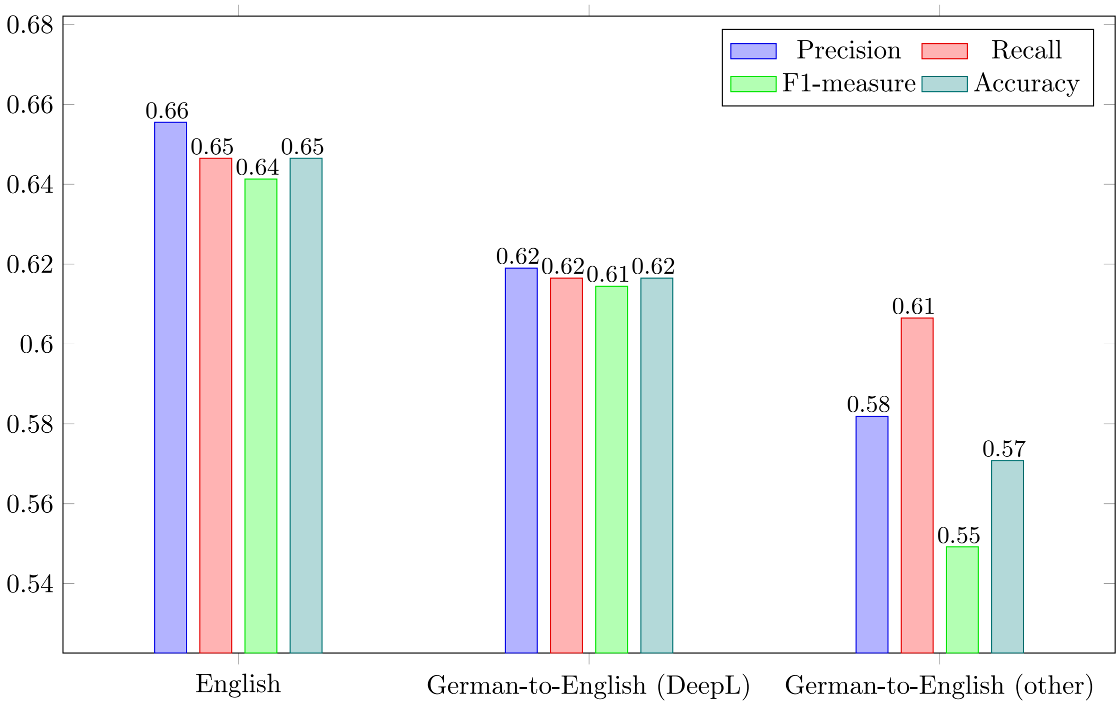

Figure 2: Evaluation results of the SVM for the English test data vs. the German-to-English DeepL translated data vs. the German-to-English differently translated data

The bar plot in Figure 2 shows the precision, recall, F1-measure, and accuracy for the English on the far left, for the DeepL translations from German to English data in the middle, as well as for the differently translated German-to-English data on the right. The y-axis shows the outcome of the evaluation measures in decimal numbers. It can be seen that all results range between 0.6 and 0.68, which are not the highest results a sentiment classifier can produce, such as the 92% accuracy in Haddi et al. (2013) or the range of 74% and 85% for translations from different languages to English in Can et al. (2018). Nonetheless, these lower results suffice since this paper investigates the effect of translation loss on sentiment analysis and not how to improve sentiment analysis.

First, the focus will be on the English vs. the German-to-English DeepL evaluation results. The precision for the English data was about 65.6%, which is about 3.7% higher than the precision of the German-to-English data with roughly 61.9%. The recall and accuracy of the English data were identical with 64.65%, for the German-to-English data both measures were the same result as well but only 61.65%, thereby resulting in a divergence of 3%, which is slightly lower than the precision difference. Lastly, the difference for the F1-measure is the lowest with roughly 2.839% as the English F1-measure is at a decimal value of 0.6413 and the German-to-English F1-measure is at 0.6245. As expected, the SVM performs better with the English data than with the German-to-English data.

Furthermore, it is also interesting to see that the differences between the metrics within each test data are varying quite a lot. For example, the precision of the English data is 0.9% higher than its corresponding recall and accuracy, while the precision and the recall of the German-to-English data diverges only by roughly 0.4%. The accuracy and recall of the English data are also 0.52% higher than the F1-measure, but the German accuracy and recall only differ by 0.2% to the German F1-measure.

The predictions also demonstrate a tendency to classify more sentence snippets as negative than positive. In all English predictions, 1237 are negative while only 763 are positive. Similarly, merely 855 sentence snippets from the German-to-English test data are classified as positive, while 1145 are classified as negative. How this trend could have happened and could maybe be fixed is a small side issue that will be investigated in section 5.

To compare the effect of translation loss further, the same experiment was performed with a different machine translation system as well. This could help answer the question of whether translation impacts sentiment because the different machine translation systems produce different translations of the same text. A pre-trained German-to-English translation model was used on the same German test data as the DeepL translator by using the Pytorch transformer pipeline. This way a direct comparison of the machine translation systems is possible, and it can be seen if the "better" translations also produce more correct predictions than the "worse" ones.

The evaluation results can be seen on the far right in Figure 2. The first and most noticeable thing are the overall lower results from the DeepL translations and the English data. The precision is at 58.19% the lowest of all three present in the figure. This is also the case for the 60.65% recall, the 54.92% F1-measure, and the 57.08% accuracy. These percentages are different from other ones, too, because they are not close together. The recall is almost as high as the DeepL recall achieved, while the accuracy is 4.6% lower than the DeepL accuracy. While the recall and the accuracy in the English and the German-to-English (DeepL) evaluation results are equal, the other translation's evaluation results vary by 3.57%. The low F1-measure stands out as well not only compared to the results from the other predictions but within the German-to-English (other) evaluation results too. It differs from the other measures between 3% to 6%. The difference between the precision and the recall is +2.5%, which is unlike the differences between precision and recall mentioned above for the other data as they have a negative difference of only up to about 1%.

Because the percentages are overall lower in contrast to the percentages of the DeepL German-to-English evaluation measures, the difference to the English percentages is greater. Besides the recall, which differs by 4% and is only 1% greater than the difference between the English and the German-to-English recall, the other performance metrics drop more than twice from the corresponding English ones, when compared to the drop of the other DeepL German-to-English performance metrics from the English performance metrics. The precision drops by 7.36%, the F1-measure by 9.21%, and the accuracy by 7.51%. If these increased differences occur due to a "worse" translation or other reasons, is explored further in the next section.

In general, it can be seen that the sentiment analyzer slightly overfits due to the higher CV result on the training data compared to the results of the test data mentioned above. Unfortunately, after trying different tuning methods, i.e. other C values and other feature matrices, this overfitting could not be reduced.

The performance of the German-to-English test data in comparison to the English data was also tested with only the use of a lexicon, in this case by using TextBlob. Table 2 shows the evaluation of the lexicon, the differences of each evaluation measure between the English and the German-to-English results, as well as the direct comparison to the SVM. The overall results are lower than the results of the SVM, which was expected considering the findings of Can et al. (2018) that using a lexicon on its own, especially on translated data, does not perform well. When comparing the English results to the German-to-English results, it is conspicuous that both have higher recall and accuracy, while the precision and F1-measure are lower by a lot. Moreover, the differences between the results are interesting, too, because the German-to-English recall, F1-measure, and accuracy are all roughly 0.6% lower than the English ones. The precision, on the other hand, is lower by 8.41% which is a big decrease. This discovery also differs from the differences of the English evaluation results compared to the German-to-English evaluation results. These are balanced in a range between –2.8% to -3.7%. The lexicon predictions are all close together because the German-to-English percentages all drop less from the

SVM results than the English percentages. However, the English precision does not decrease as much as the German-to-English precision, which causes the high variation between the two results. The 8.41% is a crucial difference between the English and German-to-English precision but also a drastic increase from the 3.7% difference of the SVM that will be discussed in the next section.
The TextBlob results of the translations of the pre-trained model, however, differed from the ones seen above and in Table 2. With the precision, recall, F1-measure, and accuracy at 0.4084, 0.4882, 0.4256, and 0.71997 respectively, a small increase compared to the DeepL evaluation results can be seen, except for the much-increased accuracy. Besides the accuracy, the evaluation of the TextBlob results for the other translations is similar to the English TextBlob performance. Yet, the other translation's accuracy is higher than the DeepL TextBlob accuracy by 23.1% and higher than the English TextBlob accuracy by 22.5%. These results are completely different from the SVM results. The SVM produces the best predictions for the English test data, then the DeepL German-to-English test data, and then it reduces its performance more for the other German-to-English test data. Also quite interesting is that the F1-measure of the other translations is higher than the English as well, even though the corresponding precision and recall are slightly under the English precision and recall.

	Precision		Recall		F1		Acc.	
	TextBlob	drop from SVM	TextBlob	drop from SVM	TextBlob	drop from SVM	TextBlob	drop from SVM
EN	0.40945	24.6%	0.495	15.15%	0.33884	30.25%	0.495	15.15%
DE/EN	0.32533	29.37%	0.489	12.75%	0.33268	28.18%	0.489	12.75%
diff. EN vs. DE/EN (TextBlob)	*-0.08412*		*-0.006*		*-0.00616*		*-0.006*	
diff. EN vs. DE/EN (SVM)	*-0,03654*		*-0.03*		*-0,02839*		*-0.03*	

Table 2: Evaluation results of the TextBlob outcomes (lexicon-based)

Finally, the sentiment analyzer's performance was also tested with another domain. The hotel test data performed similarly to the English movie review test data. The precision and recall is lower by about 2%, and the F1-measure and accuracy are lower by about 4.5% and 4.8%, respectively. These differences are comparable to the performance variation between the English movie reviews and the DeepL German-to-English test data. The precision and recall of the hotel reviews are better than the German-to-English (DeepL) ones with 63.52% and 62.595%, but the F1-measure (59.58%) and the accuracy (59.79%) are worse than the same metrics for the German-to-English (DeepL) test data.

An overview of all results and comparisons to the best performing test data can be found in the appendix of this paper in Table 4.

5 Discussion

The evaluation metrics already hint at the possibility that translation loss slightly impacts the performance of the sentiment analyzer. The higher recall for all test sets, even the translations of the pre-trained model, indicates that the sentiment analyzer can identify negative sentences quite well. The English test set, however, yields the best results in all evaluation metrics. The DeepL translations from German to English have a slight decrease in the evaluation metrics, when compared to the English ones, as explored in the results section before. The pre-trained translations produce the worst evaluation results. Especially the F1-measure is very low compared to the other test sets. This is probably due to the trouble of identifying positive sentence snippets correctly, which can be seen in the precision of 0.5819. The drop of the evaluation results from English to the German-to-English (DeepL) and then to the German-to-English (other) test sets suggest that this is due to the translations of the German test data set. In particular, the varying results for the two translations of the same dataset seem to demonstrate the effect of translation loss on sentiment analysis. In order to further verify and emphasize these findings, the annotation of the predictions is beneficial, which will be discussed in subsection 5.1.

With regards to the evaluation results of the lexicon classifications, the substantial decrease in the evaluation metrics, as well as the smaller difference between the English and the German-to-English (DeepL) results imply that the bad classification is not influenced by the translation loss, or at least not as much as the better performing sentiment analyzer built for this paper. The only unclear observation is the big difference between precisions. The lexicon seems to be better at classifying sentences as positive for English than the German-to-English (DeepL). Why this is the case is uncertain, but it could be caused by the different datasets and does not necessarily illustrate the translation loss effect here. All in all, the lexicon results show that the lexicon on its own does not perform well and cannot compare with the machine learning approach, which was mentioned in other papers before (Can et al., 2018).

Yet, what most of these papers do not include is a comparison of translation loss and quality of two different translations of the same source data. The evaluation metrics for the pre-trained model translations vary from the English and the DeepL translation results. The TextBlob performance seems better for the other translations, which achieved lower results in all other comparisons illustrated above. This is supported by the higher F1-measure that evaluates the harmonic mean of the "analyzer's" performance to correctly detect positive sentences (precision) and negative sentences (recall). TextBlob can identify both polarities better in the pre-trained model's translations than in the DeepL translations. Since this must be a result of the different translations, looking at the annotations of the translations may resolve this unexpected observation because they demonstrate that the other translations are more literal and formal than the DeepL translations.

The results of the other domain, i.e. the English hotel reviews, were rather compelling as well. The performance was analogous to the performance of the German movie reviews when translated with the DeepL translator. This could be explained by the similarity of review domains in general. Though the finer details or aspects of the reviews are more domain-specific, sentiment-carrying words, such as "great" or "horrible", are used in most cases across-domain. The smaller differences in precision and recall indicate that positive and negative hotel reviews are both similarly well classified as movie reviews (English). Thus, the sentiment analyzer often can sense the overall emotions of the review snippet correctly and can be used for multiple domains. However, the lower F1-measure and accuracy may suggest that the sentiment analyzer as a whole is not ideal for the test data, which can be expected when the SVM was trained for a different domain.

5.1 Translation Loss

Since it is the main research question of this paper, the effect of the mistranslations on the classification should be evaluated first. The annotations of all predictions give a lot of insight in this regard, such that the annotation indicates for the DeepL predictions that of all misclassifications, roughly 3.26% belonged to the misclassification category mistranslation (a).

The sentiment analyzer may have trouble correctly classifying mistranslation (a) mistakes because the leftover German words very likely do not appear in the test data and therefore the analyzer does not know to which polarity this word is a cue for. The mistranslations (b) category is not as present in the DeepL translation. Of all wrong predictions, 0.91% could be put into this category. Combining both types of mistranslations, the new percentage of mistranslations from all misclassifications is 4,17%. Overall, it seems the translation loss does not cause a lot of misclassifications, as 54.5% of sentence snippets still contain German words and 63.2% of sentence snippets with grammar or translation mistakes are correctly classified. This supports the findings of Can et al. (2018) that the translation from a low-resource language to English has an impact on the sentiment but not a striking one.

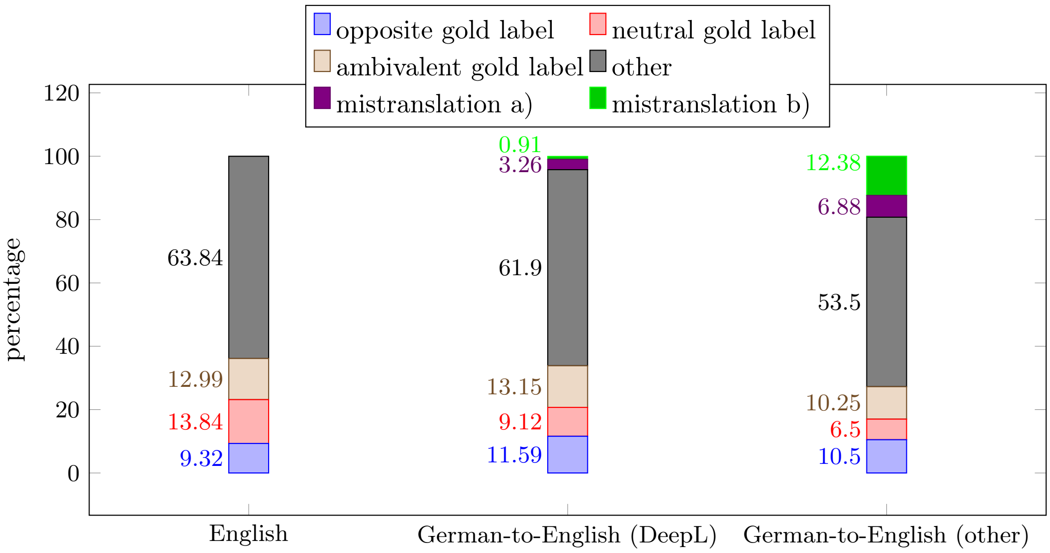

Figure 3: Distribution of the causes for all misclassifications in the English vs. the German-to-English predictions. Mistranslation a) is a translation containing German words, mistranslation b) is a translation with wrong words or grammatical mistakes

However, a closer look at the differences between the English and German-to-English predictions is necessary to back up this observation. Figure 3 shows the distribution of the possible causes for wrong predictions done by the sentiment analyzer. It contrasts the distribution for the English polarity predictions with the German-to-English (DeepL) predictions and the predictions of the pre-trained translation model (German-to-English (other)), but at first, a comparison between English and the German-to-English (DeepL) will be made.

This figure shows the (not striking) impact of the incorrect translations even better. The translation mistakes only make up 4.17% of the overall misclassifications as already observed above. Otherwise, the German-to-English (DeepL) wrong predictions are similarly distributed as the English ones. While the gold labels are not ideal, the translation loss can still be seen because both test sets have these wrong labels with about 9.322% wrong labels in the English test data and about 11.59% wrong labels in the German-to-English test data that are "correctly" identified by the SVM but counted as incorrect. The wrong gold labels probably cause the lower evaluation values compared to the other studies mentioned because these measures are calculated based on the True Positives, False Positives, etc. If these numbers are wrong, the measures are subsequently not correct. The neutral, ambivalent, and other categories are quite balanced between the two prediction sets as well because they differ by 4.725%, -0.16%, and 1,94%, respectively.

The differences, which can be seen in Figure 3, however, can also be caused by the wrong annotation on the human side because only one person, who does not have English as their mother tongue, annotated the predictions. Also, the recognition of sentiment by human annotators seems to be influenced by the machine translation as well, thus not being too dependable (Mohammad et al., 2016).

The predictions of the data translated by the pre-trained model were also annotated like the English and the DeepL translated predictions. Figure 3 also shows the subsequent distribution of the possible misclassifications compared to the English and the DeepL translations annotations. The plausible impact of the mistranslations can be evidently seen. The percentage of all mistranslations (a) that probably cause the misclassifications doubled for the other translation. Interestingly, the annotations showed that most of the mistranslations (a) in the DeepL translations are correct or nearly correct in the other translations. Since these were potentially caused by the misspellings which DeepL then could not handle, it could be assumed that the

other translation model can deal better with such spelling errors, though this cannot be said generally.
Another interesting observation that occurs in the DeepL translations and in the translations of the pre-trained machine translation model is how the model deals with sayings, metaphors, or idioms. The German idiom *"mit etwas/jemanden nichts am Hut haben"*, for example, expresses that you have no interest in something or someone. Both translators were not able to correctly translate this phrase used in the sentence:

(1) *"Wär den Film schlecht findet hat ihn einfach nicht verstanden oder mit dem Thema* ***nix am hut*** *die Insider wissen was mit dem Film gemeint ist und was damit kritisirt wird!"*

(2) *"Who thinks the movie is bad simply did not understand it or* ***has nothing to do with*** *the topic insiders know what the movie stands for and what it criticises!"*

The DeepL translation looks like this:

1.a *"wär the film badly find ha him simply not understood or with the topic* ***nix am hut*** *the insider know what is meant by the film and what is criticisirt with it!"*

The pre-trained translation model, however, built this sentence:

1.b *"if the film doesn't like it, it just didn't understand it or with the theme* ***nix am hat*** *the insider know what is meant by the film and what is being criticized by it!"*

The pre-trained translation model translated the sentence word-by-word and thus was able to translate the misspelled words "wär[e]" → "would", which should be "wer" → "who" and "kritisirt" → "kritisiert" → "criticized". It also attempted to translate the idiom, but only managed the word "hut" → "hat". This can also be due to the colloquial form of "nichts"(nothing") → "nix". However, the DeepL translator, as seen before, struggles with misspellings, hence it only translated the words and phrases it knew and completely ignored the misspelled words, as well as the idiom.
During the annotation, the increased number of mistranslations (b), i.e. grammar mistakes and wrong words, stuck out as well. While the predictions of the DeepL translation only have 7 sentences, which are misclassified and are annotated as mistranslation (b), 99 of all misclassifications in the predictions of the other translation model may be caused by grammar mistakes or words that were translated incorrectly. On the one hand, it is important to keep in mind that the category mistranslations (b) is highly subjective and may contain more annotation mistakes than the mistranslations (a). Then again, the other translation model mostly translated very literally and thus the translated sentence snippets are often genuinely grammatically incorrect. For instance, *"[..]these 30 minutes did not address me at all."* is the literal translation of *"[..]diese 30 Minuten haben mich überhaupt nicht angesprochen."*. "Angesprochen" can mean *"A hat B angesprochen"* which can have the sense *"A addressed B"*, but also of *"A appealed to B"*, which is the case in this context.
The problem of choosing the correct word if there are multiple according to the context is not only a problem of ambiguity, which will be briefly examined later on but - for other cases - a problem of loss of lexical richness and diversity as explained by Vanmassenhove et al. (2019). A lot of cases that apply to their findings can be detected.
As briefly mentioned in the background section of this paper, Vanmassenhove et al. (2019) showed that machine translation systems tend to generalize the translations such that the translation often contains the most frequent words and is not considered to be rich in lexical diversity. In the example given above, it could be the case too that the most frequent word was chosen. When looking at the lemmatized list from "BNC database and word frequency lists" ordered by frequency by Kilgarriff (1995), the verb "address" shows up 5872 times in the 100M-word corpus BNC, while the verb "appeal" only appears 4764 times in the same corpus. Thus, the pre-trained model's choice to select "address" over "appeal" is quite comprehensible.
The literal translations may also clarify the quite unexpected results of TextBlob for the pre-trained model's translations. The words, which were chosen by the model, often seem to be the most frequent words. TextBlob, thus, may have more sentiment values for the other translations

as they contain the more commonly used words compared to the DeepL translations. Although this does not explain the better results compared to the English TextBlob evaluation results, it may resolve some questions raised, when looking at the evaluation metrics for the TextBlob results of the other translation.

In addition to the grammar mistakes and choosing the more frequent words in ambiguous cases, incorrectly selected words appeared more often compared to the DeepL translations, such as *"hello i can not **sit** differently: the film is a classic."*, which should be *"hello i can not **say** differently: the film is a classic."*. At the same time, the DeepL translation of this sentence does not only choose the correct verb but seems to be better formed, too: *"hello i can't **say it** any other way: the film is a classic."*. The sentiment analyzer is able to correctly classify the DeepL translation as a positive sentence, while the other translation is misclassified as a negative sentence. Along with these problems, the mistranslations sometimes caused a complete polarity shift, too. The sentence *"Ich habe zuvor kein Buch von Sherlock Holmes gelesen und auch mich nicht weiter mit dieser Person beschäftigt."* is translated by DeepL as

(a) *"i have not read any of sherlock holmes' book before, nor have i delved further into this person."*

The pre-trained model translates the sentence as follows

(b) *"i did not read a book by sherlock holmes before and did not continue to deal with this person."*

Even though the sentence in German is, subjectively, considered a neutral phrase, maybe with a slight tendency to negative, the translation by the pre-trained model can be, again subjectively, classified as negative. The DeepL translation is able to capture the neutral/negative sentiment better than the pre-trained model translation. This example did not cause a misclassification in the predictions. However, these shifts should be kept in mind if different test data is used.

Ultimately, seeing the many issues in the translations by the pre-trained model, the immense difference between the number of mistranslations (b) seems logical.

Altogether it seems the DeepL translator has more "real-world" knowledge as it is able to translate movie names correctly and adapt to the rather colloquial circumstances of the sentences as part of movie reviews than the pre-trained machine translation model. This would also explain the better performance of the sentiment analyzer with the DeepL test data.

All of the results stated above emphasize the findings of other studies such as the work of Mohammad et al. (2016) or Can et al. (2018), that the machine-translated test data produced competitive polarity predictions in comparison to English test data. As already explored above, some misclassifications can be caused by words that are still in the input language and cannot be processed properly by the sentiment analyzer. Interestingly though, the translation loss percentages given by Can et al. (2018) vary highly from the percentages seen above. Figure 4 illustrates the contrast quite clearly. While for the German-to-English translations, both by DeepL and the pre-trained model, under 10% of all misclassifications are mistranslations (a), the percentages presented by Can et al. (2018) range from 10.7% up to 24.9% for the same "category", as they only focussed on sentences which still contained words in the source langugage. These variations can be explained by different translation systems again. Can et al. (2018) translated their low-resource language test data sets with the Google Translate API and not with the DeepL Translate API. This is the confirmed result of DeepL creating better translations and a newer experiment supported previous indications of other researchers in this direction, too.

Macketanz et al. (2021) used their evaluation framework TQ-AutoTest on both machine translation systems and another one called Lucy, which is not of importance in this context. The framework evaluates an automatic test suite containing various phenomena categories and phenomena that classify about 5000 sentence segments which are taken from different sources such as bilingual corpora (Macketanz et al., 2021). With the help of regular expressions, TQ-AutoTest can judge whether an output of a machine translation system is correct or incorrect (Macketanz et al., 2021). These judgments exhibit the better performance of DeepL compared to Google in all phenomena categories except subordination, which describes the structure of a sentence,

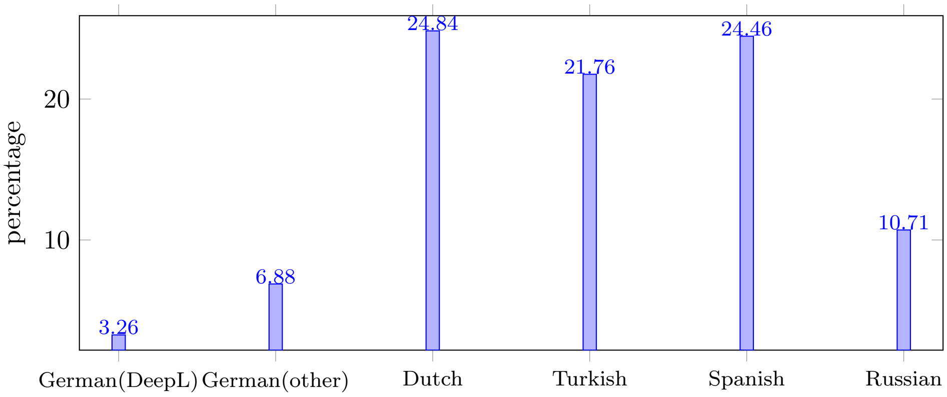

Figure 4: Comparison of the mistranslation/misclassification percentages from this paper and from Can et al. (2018)

where the subclause is dependent on the main clause[8] (Macketanz et al., 2021). Besides this verification regarding DeepL's performance in contrast to Google's, the experiment points out DeepL's ability to recognize colloquial German and translate it into colloquial English (Macketanz et al., 2021). A similar observation was made in this thesis earlier, when comparing the DeepL translations with the translations by the pre-trained model and is thus strengthened in this regard.

Nevertheless, it is crucial to keep the different annotations in mind, when comparing the translation loss as seen in Figure 4. The annotation of Can et al. (2018) might be more proper or have other criteria than only counting non-English words. Also, the ratio of the mistranslations to the misclassifications can vary and cause these differences between the percentages of Can et al. (2018) and the ones of this paper. The performance of their sentiment analyzer is better than the SVM presented here. If there are fewer misclassifications, the misclassifications containing mistranslations weigh more and thus the percentage is higher. The SVM here makes more false predictions and thus the category, mistranslations (a), is smaller compared to the other possible cases seen in Figure 3. Since the actual ratios of the mistranslations from Can et al. (2018) are not public, this comparison is still included as it might contain some noteworthy arguments.

A rather new observation is the impact of the loss of lexical diversity on sentiment analysis. Vanmassenhove et al. (2019) stated the importance of lexical richness for data mining tasks but did not investigate any further, especially in regard to the effect this kind of translation loss has on the sentiment of the sentence. Balahur and Turchi (2012) pointed out that the quality of the translation also has an impact on the performance of the sentiment analyzer. However, these results were based on a study using translations from English to Spanish, French, and German as training and test data. Thus, the findings presented here in this paper support the findings ofBalahur and Turchi (2012), when using translated data only as of the test data and not the training data.

Furthermore, what was not explored more as persistent in this thesis so far but briefly mentioned by Mohammad et al. (2016) are semantic phenomena like ambiguity, metaphors, and sarcasm. These can flip the polarity but are very difficult to detect for machine learning approaches. Sarcasm, for example, is an ongoing issue that is investigated by a lot of researchers mostly separate from sentiment analysis in other NLP in sentiment analysis but in sentiment analysis as well (Godara and Aron, 2021). The problem with automatic sarcasm detection is the topic specificity of the context as well as other "metadata" such as the culture or people related to the statement, which the machine is not capable to apply in these situations (Maynard and Greenwood, 2014). Maynard and Greenwood (2014) attempted to improve sentiment analysis by detecting sarcasm first, which proved to be beneficial but did not "perfect" the sentiment analysis. However, for sarcasm detection, a clear definition of what counts as sarcasm and what does not is needed, which is a challenging task on its own. For instance, the study mentioned above chose to define

[8] https://en-academic.com/dic.nsf/enwiki/5924719

the use of sarcasm, when a person intends to express the opposite of what was said to either amuse or affront others, but they make it clear that sarcasm does not always flip the polarity (Maynard and Greenwood, 2014).

Ambiguity, like sarcasm, proves to be more complex to solve in NLP. Single words can have more than one meaning that varies with the context, though a phrase can be ambiguous based on the context as well. The ambivalent category, for example, contains a few ambiguous sentences. There are few studies dealing with ambiguity and its effect on sentiment analysis. Deng et al. (2017) claim that dealing with ambiguity in sentiment analysis works better with pairwise dependency parsing instead of using phrase-level sentiment analysis approaches. They were able to see improvement of the sentiment analysis but also see their study as a direction for future works because the results are limited by the parsing accuracy (Deng et al., 2017).

The training and test data sets of this thesis did not seem to contain any metaphors. Instead, they contained some idioms, which in the German-to-English test data caused some trouble in the translation (and thus in the classification). A study with English idioms was carried out by Williams et al. (2015), which resulted in an increase of the evaluation measures by 20 percent points and was achieved by using an idiom feature in the sentiment analyzer.

So there are already possibilities out there for dealing with these problems, although only for one problem at a time and not combined, which is necessary because all phenomena often appear in the data at once and not each one separately. Also, until now all of these phenomena are mainly investigated for English, which poses problems for low-resource languages such as German (Godara and Aron, 2021). Considering the translation of German idioms to English and then identifying the sentiment of the idioms seems like a different challenge.

5.2 Balanced Training Data

Another observation made in the results was the big differences within the English evaluation outcomes and the slightly smaller differences within the German evaluation results. If one looks at the distribution of the classes for both predictions of the data test data sets, as was also done in section 4, this result makes more sense. The classes are a little bit more "balanced" for the German-to-English predictions in contrast to the class distribution of the English predictions since the negative and positive predictions differ by -290 for the German-to-English predictions and by -474 for the English predictions. This may lead to the conclusion that the lower the evaluation metrics are and the more balanced the predictions, the closer the metric outcomes are to each other.

	Precision	Recall	F1	Accuracy
balanced EN	~0.6563	0.647	~0.6417	0.647
diff. balanced vs. unbalanced (EN)	*-0.0009*	*-0.0005*	*-0.00038*	*-0.0005*
unbalanced EN	~0.65554	0.6465	~0.6413	0.6465
diff. balanced EN vs. balanced DE/EN	**-0.03181**	**-0.0255**	**-0.0225**	**-0.0255**
diff. unbalanced EN vs. unbalanced DE/EN	**-0,03654**	**-0.03**	**-0,0284**	**-0.03**
balanced DE/EN	~0.62449	~0.6215	~0.6192	0.6215
diff. balanced vs. unbalanced (DE/EN)	*-0.00549*	*-0.0027*	*-0.0047*	*-0.0027*
unbalanced DE/EN	~0.619	0.6165	~0.6145	0.6165

Table 3: Evaluation results, when using balanced training data. Draws comparison to evaluation results, when using unbalanced training data

Overall, this also shows that the sentiment analyzer tends to classify more sentences as negative than positive. For example, the sentence snippet "please do not miss the opportunity to see this on the big screen." from the English data is misclassified as negative, though it actually is positive. The negative tendency could be the consequence of unbalanced training data with more negative reviews than positive (5326 neg. vs. 5242 pos.) as was the case for Can et al. (2018). Whether the unbalanced data is the true cause of this, was tested by reducing the negative training data to 5242 so it is equal to the positive training data. Table 3 shows all the evaluation results for the English test data and the German test data, when the training data is balanced (balanced EN, balanced DE/EN) and when it is unbalanced (unbalanced EN, unbalanced DE/EN) which was investigated before. It also presents the respective differences between the balanced and the unbalanced data for each test data set (in italics), as well as between the two data sets (in bold). The differences were calculated by subtracting the balanced results from the unbalanced results and subtracting the English results from the German-to-English results. It can be seen that the German-to-English results improved more when using balanced training data than the English results. More interesting, however, is the decrease of the difference between the evaluation results of the English (balanced training data) and the German-to-English (balanced training data). A reduction of roughly 0.5% is noticeable in all differences between the evaluation measures. The German-to-English accuracy and recall only differ from the English accuracy and recall by -2.55% instead of 3%, while the precision varies just by -3.181% as opposed to the 3.654% when using unbalanced training data.

Finally, the F1-measure is also reduced from 2.839% to only 2.247%. These results support the hypothesis that balanced training data results in more balanced predictions (Can et al., 2018). Nevertheless, this should be independent of the translation loss aspect and probably mainly influences the ambivalent category or other misclassifications (see Figure3). Also, one has to consider that the test data, English and German-to-English, often have wrong labels, which indicates that looking at the class distributions as done above, is not ideal. Instead, the annotations should be looked at to see how many of all negative predictions and of all positive predictions have a wrong gold label.

6 Conclusion

This study investigated the impact of translation loss on sentiment analysis with a SVM sentiment analyzer and different machine translation systems that translated German movie reviews to English.

Taking everything discussed so far into account, the presented study supports the findings of previous research. The machine translation of low-resource languages, here German, to English and using it as test data for a sentiment analyzer trained on English data is feasible. Moreover, the importance of the quality of the translation is demonstrated. The more accurate translations yielded better polarity classifications that were closer to the English sentiment predictions than the more literal translations with more German words in the sentences and many ungrammatical sentences. This also supports the findings of Balahur and Turchi (2012) that the quality of the translations impacts the performance of the sentiment classification. Thus, one could conclude that the bigger the translation loss, the bigger the influence on the sentiment classification.

The findings of this paper could be backed up with further explorations of this observation. In particular, the annotation have to be more accurate than in this experiment. Instead of one annotator, who does not have English as their mother tongue, multiple native English speakers should evaluate whether a sentence is grammatical or not and if the translation appears to contain the correct words. Conceivably, the annotators should have no or only a little knowledge about the source language, especially the grammar. That way, bias can be reduced to the minimum. With regards to mistranslations (a), the annotation should still hold as the identification of German words in the translated sentence snippets will most likely be best identified by a native speaker.

Additionally, the observed phenomena, sarcasm, ambiguity, and idioms are still in need of applications for languages other than English. German idioms, for instance, were not translated properly and thus not correctly classified. If the translations of the idioms are always the same, a dictionary for the idiom and its corresponding sentiment could be built. This dictionary could then maybe be added as a feature for the SVM and the performance differences of the SVM with the idiom dictionary and of the SVM without the idiom dictionary may be investigated.

Furthermore, these findings should also be confirmed with sentiment intensity classification. Because Vanmassenhove et al. (2019) already established the loss of lexical diversity in translation as basic scientific research and because its effect on polarity sentiment analysis is hinted at by this study, further studies could explore this based on sentiment intensity analysis. This could be an interesting experiment, as the varying target words that were chosen for the same source word by different machine translation systems could also vary in their intensity and the translation would affect the sentiment analysis even more. For example, while annotating for this research, DeepL and the pre-trained model, selected different English words - "strange" and "alien" - for the original German word "fremd", which may cause a slight shift in the intensity of the sentence. Also, it could be observed that the pre-trained model sometimes left out intensifiers, such as "very", which are important for sentiment intensity analysis. Therefore it is a fascinating possible further research question to investigate whether the quality of the translation has an impact on sentiment intensity and if it does, to see whether it is more striking than the one for sentiment polarity.

When using the equal amount of positive and negative reviews as training data, some important observations could be made, hence making the balancing of training data a possible improvement to decrease the difference between the evaluation measures of the English predictions and the German-to-English predictions. The findings could be explored a bit more in a further study to see where the increase of the predictions especially for German-to-English test data came from. Even though the gold labels and perhaps even the test data itself are not ideal for sentence-level sentiment analysis, they may be good for document-based sentiment analysis. Nevertheless, it could be the case that document-level analysis is more difficult to perform well in comparison to sentence-level analysis due to the mix of different sentiments within one document. This may cause confusion in the sentiment analyzer as it is given a lot of clues hinting to several sentiments. Despite these assumptions, exploring the document-level sentiment analysis could be done in the future. Another possible enhancement to this is first classifying the sentences as either objective or subjective and then classifying the subjective sentences into positive and negative to reduce misclassifications because some sentence snippets are summaries of the movie or other facts.

In conclusion, the investigation of sentiment analysis still has, despite much research already, much to look into and still is a topical issue to be further discussed. However, as this study and others show, machine translations of low-resource languages can be used for NLP tasks, such as sentiment analysis, without causing too many troubles.

References

Ahmad, M. and S. Aftab (2017). Analyzing the performance of svm for polarity detection with different datasets. *International Journal of Modern Education & Computer Science 9*(10).

Bahrainian, S.-A. and A. Dengel (2013). Sentiment analysis using sentiment features. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, Volume 3, pp. 26–29. IEEE.

Balahur, A. and M. Turchi (2012, jul). Multilingual sentiment analysis using machine translation? In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, Jeju, Korea, pp. 52–60. Association for Computational Linguistics.

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

Can, E. F., A. Ezen-Can, and F. Can (2018). Multilingual sentiment analysis: An rnn-based framework for limited data. *arXiv preprint arXiv:1806.04511*.

Deng, S., A. P. Sinha, and H. Zhao (2017). Resolving ambiguity in sentiment classification: The role of dependency features. *ACM Transactions on Management Information Systems (TMIS) 8*(2-3), 1–13.

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM 56*(4), 82–89.

Godara, J. and R. Aron (2021). Sarcasm detection on social network: A review. *Annals of the Romanian Society for Cell Biology 25*(6), 3761–3771.

Guhr, O., A.-K. Schumann, F. Bahrmann, and H.-J. Böhme (2020, May). Broad-Coverage German Sentiment Classification Model and Dataset for Dialog Systems. This repository contains the trained models as well as the training data.

Haddi, E., X. Liu, and Y. Shi (2013). The role of text pre-processing in sentiment analysis. *Procedia Computer Science 17*, 26–32.

Hastie, T., J. Friedman, and R. Tibshirani (2009). *The elements of Statistical Learning: Data Mining, Inference, and prediction* (2. ed. ed.). Springer.

Hutto, C. and E. Gilbert (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, Volume 8.

Jurafsky, D. and J. H. Martin (2021). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition, 3. ed. (draft of december 29, 2021.). Available from: `https://web.stanford.edu/~jurafsky/slp3/`, Accessed: 2022-2-10.

Kilgarriff, A. (1995). Bnc word frequency lists. `http://www.kilgarriff.co.uk/bnc-readme.html`. Accessed: 2022-2-20.

Loria, S. (n.d.). Textblob: Simplified text processing — textblob 0.16.0 documentation. `https://textblob.readthedocs.io/en/dev/`. Accessed: 2022-2-10.

Maas, A. L., R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, USA, pp. 142–150. Association for Computational Linguistics.

Macketanz, V., A. Burchardt, and H. Uszkoreit (2021). Tq-autotest: Novel analytical quality measure confirms that deepl is better than google translate. *nd*.

Maynard, D. G. and M. A. Greenwood (2014). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec 2014 proceedings*. ELRA.

Medhat, W., A. Hassan, and H. Korashy (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal 5*(4), 1093–1113.

Mohammad, S. M., M. Salameh, and S. Kiritchenko (2016). How translation alters sentiment. *Journal of Artificial Intelligence Research 55*, 95–130.

Pang, B. and L. Lee (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *arXiv preprint cs/0409058*.

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

Schmidt, T., J. Dangel, and C. Wolff (2021). Senttext: A tool for lexicon-based sentiment analysis in digital humanities.

Tian, L., C. Lai, and J. D. Moore (2018). Polarity and intensity: the two aspects of sentiment analysis. *arXiv preprint arXiv:1807.01466*.

Tymann, K., M. Lutz, P. Palsbröker, and C. Gips (2019). Gervader-a german adaptation of the vader sentiment analysis tool for social media texts. In *LWDA*, pp. 178–189.

Vanmassenhove, E., D. Shterionov, and A. Way (2019). Lost in translation: Loss and decay of linguistic richness in machine translation. *arXiv preprint arXiv:1906.12068*.

Waltinger, U. (2010). Germanpolarityclues: A lexical resource for german sentiment analysis. In *LREC*, pp. 1638–1642. Citeseer.

Williams, L., C. Bannister, M. Arribas-Ayllon, A. Preece, and I. Spasić (2015). The role of idioms in sentiment analysis. *Expert Systems with Applications 42*(21), 7375–7385.

Appendix

	SVM EN	**SVM DE-EN (DeepL)**	**SVM DE-EN (other)**	**SVM EN (hotel reviews)**
Precision	65.554%	61.9%	58.19%	63.52%
Precision: Difference to SVM (EN)	-	-3.654%	-7.364%	-2.034%
Recall	64.65%	61.65%	60.65%	62.595%
Recall: Difference to SVM (EN)	-	-3.0%	-4.0%	-2.942%
F1-measure	64.13%	61.45%	54.92%	59.58%
F1-measure: Difference to SVM (EN)	-	-2.68%	-9.21%	-4.55%
Accuracy	64.65%	61.65%	57.08%	59.79%
Accuracy: Difference to SVM (EN)	-	-3.0%	-7.57%	-4.86%

	TextBlob EN	**TextBlob DE-EN (DeepL)**	**TextBlob DE-EN (other)**	**TextBlob EN (hotel reviews)**
Precision	40.95%	32.53%	40.84%	30.02%
Precision: Difference to SVM (EN)	-24.604%	-33,024%	-24.714%	-35.534%
Recall	49.5%	48.9%	48.82%	48.32%
Recall: Difference to SVM (EN)	-15.15%	-15,75%	-15.83%	-16.33%
F1-measure	33.884%	33.268%	42.56%	36.07%
F1-measure: Difference to SVM (EN)	-30.246%	-30.862%	-21.57%	-28.06%
Accuracy	49.5%	48.9%	71.997%	56.13%
Accuracy: Difference to SVM (EN)	-15.15%	-15.75%	**+7,347%**	-8.52%

Table 4: Overview of all evaluation results using the results of the SVM for the English test data as a comparison "baseline"