Automatic Detection of Different Types of Translation Based on Translationese Features

By:

Yu-Wen Chen

Supervisors:

Dr. Antonio Toral

Dr. Cristina España-Bonet

Prof. Dr. Josef van Genabith

Master thesis

European Masters Program Language & Communication Technologies

Rijksuniversiteit Groningen & Universität des Saarlandes

December 24, 2020

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Ich versichere, dass die gedruckte und die elektronische Version der Masterarbeit inhaltlich übereinstimmen.

Declaration

I hereby confirm that the thesis presented here is my own work, with all assistance acknowledged. I assure that the electronic version is identical in content to the printed version of the Master's thesis.

Groningen, December 24, 2020
Signature

ACKNOWLEDGEMENT

I would like to first thank my main supervisor, Dr. Antonio Toral, for inspiring my interest in the field of translationese. I am grateful for his knowledgeable guidance when I am confronted with difficulties and his support throughout each stage of the process.

I would like to thank my co-supervisors in the Saarland University, Dr. Cristina España-Bonet and Prof. Dr. Josef van Genabith for organizing monthly translationese meeting and their invaluable feedback which certainly helped to improve the work.

I would also like to thank the Center for Information Technology of the University of Groningen for their support and for providing access to the Peregrine high performance computing cluster. I would particularly like to single out Cristian, Klemen, and Fokke for their prompt help with the technical issues.

The past two and half years pursuing European Masters program in Language and Communication Technologies (EMLCT) had been the most challenging time in my life. I am thankful for everyone who participated in this journey and made me believe I can do it.

I would like to acknowledge EMLCT coordinators, Dr. Gosse Bouma and Dr. Jürgen Trouvain, for helping me with course selection and the administration work for the program.

Besides, I would also like to express my eternal gratitude to my family for their support and I know they are always there for me. Finally, I want to show my deepest appreciation to Michiel Timmermans for his endless support and encouragement.

ABSTRACT

Studies have shown that readers still prefer human translation (HT), even over translations produced by state-of-the-art Machine Translation systems. Often overlooked when measuring translation quality in machine translation (MT) as compared to HT are lexical and syntactic differences. This paper studies the aspects of *translationese* that are valuable for distinguishing different translation types and whether such unique phenomena can be detected by machine-learning classifiers. In this study, translationese features are defined under the framework of *translation universals* in four aspects - simplification, normalization, explicitation, and interference. We expect that machine-based translation reveals more pronounced translationese than HT. A Support Vector Machine (SVM) classifier is built to conduct three binary classifications of the three commonly available translation types - MT, HT, and post-edit (PE). The results suggest that machine-based translation (i.e., MT and PE) exhibits translationese characteristics which are less present in HT. It is advised that future research should incorporate deeper linguistic representations into the features. Finally, when making cross-comparisons of translation types in future, it is also advised that a single dataset consisting of the same source texts is used, alongside machine-based translations generated from one fixed MT system.

CONTENTS

1 INTRODUCTION

When reading translated texts, it is a common feeling for native language readers to spot the 'unnaturalness' compared with texts written in the original languages. The linguistic phenomenon of this 'unnaturalness' in the translated texts is termed as *Translationese* (Gellerstam, 1986) or *third code* (Frawley, 1984). This difference between original and translated texts does not indicate inferior quality. It is even preferred in some situations than pure target language form (Toury, 1979). This is no surprise since human translators might take the reader's sociocultural background into account. Translated texts are made syntactically simpler or more explicit than source texts by human translators to fit the target cultural and linguistic systems.

Meanwhile, as the development of machine translation (MT) systems rapidly evolves, translations are no longer merely hand-crafted by human translators. MT systems are actively involved as part of the translation process to help human translators in productivity (Plitt and Masselot, 2010). On the other hand, some professional translators still disfavour the use of MT in many respects (Bundgaard, 2017), one reason being the 'inadequate translation' produced by MT systems. This gap between human-quality translation and actual MT has been to be distinguishable by machines in terms of lexical usage (Carter and Inkpen, 2012), grammaticality and fluency (Arase and Zhou, 2013; Chae and Nenkova, 2009), and coherence (Nguyen-Son et al., 2019).

Despite the difference, some neural MT systems have claimed to reach *human parity* (Hassan et al., 2018), that is, there is no significant difference between the quality scores of MT and HT. Toral et al. (2018a) further reassessed this claim, which shows MT is still significantly different from HT when translating the texts written in the original language according to human's direct assessment (DA). Besides, human assessors still show a strong preference towards HT compared with MT when evaluating the whole document instead of an isolated sentence (Läubli et al., 2018). Nonetheless, in the news translation shared task at WMT19 (Barrault et al., 2019), one MT system used for translation direction of English to German reached human parity, but the super-human performance was not yet achieved (Toral, 2020).

While most of the MTs still struggle to reach human quality, research has shown that the quality of post-edits (PEs) is equivalent to that of HTs (Garcia, 2010) or even better (Green et al., 2013). Post-editing is a common translation workflow conducted after MT systems translate the source texts. During post-editing, human translators annotate and edit the errors made by MT systems to produce the final translation. Due to this procedure's nature, however, the final translations (i.e., PEs) are primed by the raw MT outputs (Green et al., 2013). As a result, PEs are found to exhibit 'exacerbated translationese', termed as *post-editese* (Toral, 2019). PEs have lower lexical variety and density, suggesting that they are lexically simpler than HT. Moreover, PEs also have more interference from the source language than HT. In sum, we could illustrate that these three currently popular translation types all exhibit translationese in different forms that stand apart given the different challenges they are facing.

Such distinctive features of the respective translation types can be effectively identified by the machine-learning algorithms. The linguistic difference between originals and HT has already been demonstrated to be machine-learnable by several studies. A more systematic approach by Volansky et al. (2013) incorporates the notion of translationese hypotheses as the inspiration for the features used in the classification task of originals and HT. These translationese features are grouped into four categories under the framework of *translation universals*, namely simplification, explicitation (Baker et al., 1993), normalization, and interference (Toury, 1995). Simplification refers to using simpler syntax and lexicon; explicitation means the implicity in the original texts is made explicit in the translation; normalization indicates the standardized texts with conventional grammar, and interference implies that the source language's footprint remains in the translation. It shows among all hypotheses, features that model the phenomenon of *interference* from the source texts are the best performing features for distinguishing between originals and HT. Likewise, the linguistic traits between MT and HT is machine learnable and detectable. Carter and Inkpen (2012) use unigram features with Support Vector Machines (SVMs) classifiers to distinguish between MT and HT, resulting an accuracy of 99.8%. Arase and Zhou (2013) also employs and SVM classifier with features modeling the sentence's fluency and grammaticality, achieving an accuracy of 95.8% in the detection of MT and HT.

Following this line, we hope to extend the research scope to different translation types that also exhibit unique translationese phenomena. We expect the research on the translationese characteristics of commonly available translation types could shed light on the deficiencies of the MT systems (Lembersky et al., 2011) and post-editing strategy. Inspired by Volansky et al. (2013), we adopt features under the framework of *translation universals* and employ machine-learning classifiers to investigate which translationese traits are indicative of different types of translation. We aim to answer the following questions:

RQ1. **Can human translation be distinguished from machine-based translation according to translationese features?** Under the framework of translation universals, whether *human* translationese exhibits distinguishable phenomena from *machine* translationese (i.e., MT and PE). We hypothesize that machine-based translations will exhibit more-translationese like phenomena than human translation. Regarding the classification accuracy, we also hypothesize that if the MT system's translation quality increases, the classification accuracy will decrease, meaning it is harder for the classifier to learn the difference.

RQ2. **If the answer to RQ 1 is yes, then which characteristics of translationese are valuable to distinguish human and machine-based translation?** That is, what is the machine-learnable difference between *human* translationese and *machine* translationese?

RQ3. If the answer to RQ1 is yes, then MT and PE are both distinguishable from human translation. **However, between these two machine-based translations, what are the most useful characteristics to distinguish the two?** We hypothesize that PE manifests less severe translationese than MT. Since PE is primed by MT but with an additional human touch, what machine-detectable characteristics does the human editor erase from or add into the MT texts?

The rest of the thesis is organized as follows. In chapter 2, we provide a brief introduction about the previous studies on the thesis's fundamental concepts. In chapter 3, we describe the datasets used for our experiments. The definition and calculation of translationese features and the experiment scheme is presented in chapter 4. In chapter 5, we present the experiments' results and provide the analysis and interpretation of the feature importance. Finally, in chapter 6, we state the answers to our research questions and conclude our findings.

2 RELATED WORK

This chapter will present the previous research regarding the fundamental concepts of this thesis. In section 2.1, we introduce the development and characteristics of three types of translation used in our experiment - MT, HT, and PE. Section 2.2 reviews the studies of unique phenomena exhibited in the translated texts - *translationese* and *post-editese*. Finally, section 2.3 gives an overview of the studies using computational approaches for classification tasks between different types of translations.

2.1 DIFFERENT TYPES OF TRANSLATION

Translation, serving as an intermediate between original and target languages, is commonly depicted as the work produced by bilingual and professional translators with specific domain knowledge. HTs are often not just merely translated texts. Instead, they are adapted to the target languages' linguistic norms and take the reader's knowledge into account (Ahrenberg, 2017). The native language and translation expertise of the human translator also impact the characteristics of the resulting translated texts. A recent study (Popovic, 2020) has shown that the translator's expertise influences sentence length, lexical and morpho-syntactic variety. Professional HT has a longer sentence length and the greatest lexical variety in the translation direction from German to French. Word length and lexical variety are also higher when translating German into the translator's native (French) language.

Such HTs were the center of translation studies until the wide availability of online machine MT applications (Koponen, 2016). The MT applications benefit the public and are commonly employed to translate short life-cycle digital texts such as reviews, online documentation, and social media content without human effort to intervene (Moorkens et al., 2018). Moreover, involving MT in the translation procedure has shown to be more productive than translating from scratch in the technical (Plitt and Masselot, 2010) and news (Martín and Serra, 2014) domain. Nevertheless, MT still exhibits different linguistic attributes than HT. Ahrenberg (2017) compares human-translated Swedish and machine-translated Swedish generated by Google Translate, and it has shown that MT is more similar to the source texts than HT in terms of length, information flow, and structure. It also reveals that HT takes different procedures such as sentence splitting and paraphrasing that seem to be out of MT's reach. These procedures notably share the purpose of making the translated texts more compatible with a fluent reader's experience.

Among all the domains where MT is actively used, news text has received particular interest due to its daily demand for high productivity and commercial value. In this specific domain, state-of-the-art MT systems have claimed to achieve *human parity* (Barrault et al., 2019, section 3.8) for three language directions (DE ↔ EN and EN → RU), that is, human assessors perceive the quality of MT as equivalent with HT. There is no significant difference between the quality scores attained by MT and HT. Hassan et al. (2018) also claim the MT translated from Chinese into English has achieved human parity. Toral et al. (2018a) reassess the human parity in Hassan et al. (2018) and find out that HT still significantly outperforms MT when translating source texts written in the original language.

Despite the characteristic difference between HT and MT, with the development of machine translation paradigms shifting from the phrase- and rule-based MT (PBMT and RBMT) to neural MT (NMT), the MT's quality has increased. Toral and Sánchez-Cartagena (2017) compare the nine translation direction outputs from best PBMT and NMT submitted to the WMT16 translation task and conclude that NMT systems produce more fluent texts than PBMT and the NMT's reorderings are closer to the reorderings of the human reference than PBMT.

Regardless of the wide application and human parity claim, MT for various language pairs and domains are still far from published quality. Post-editing (PE) (i.e., annotating the errors) the raw MT has thus integrated into the workflow of professional translators as a common practice to improve the translation quality and productivity (O'Brien et al., 2014; Koponen, 2016). Toral et al. (2018b) study the effect of post-editing along three dimensions: temporal, technical, and cognitive effort. It is found that post-editing the NMT output increases translation productivity by 36 % compared to translating from scratch. Although MT primes PE, research has shown that PE texts have equivalent or even better quality than HT from scratch judged by human accessors (Green et al., 2013; Koponen, 2016; Daems et al., 2017).

2.2 TRANSLATIONESE AND POST-EDITESE

Translated texts have proved to be syntactically and lexically different from original non-translated texts. This phenomenon, termed as *translationese*, is first discussed by Gellerstam (1986) in which compares original Swedish and translated Swedish from English. He suggests that translationese does not necessarily mean inferior translation quality; instead, it is an empirical linguistic phenomenon exhibited only in the translated texts. Translation thus stands along as a *third code* (Frawley, 1984), meaning it is a sub-language of each language involved. Moreover, such distinct features are claimed to be universal (Baker et al., 1993; Baker, 1995): the translated texts have shown this specific feature distribution regardless of the source languages. These translation universals are grouped into four categories: simplification, explicitation, (Baker et al., 1993) normalization, and interference (Toury, 1995).

The simplification hypothesis indicates that translated texts are simpler lexically and syntactically than originals. Blum and Levenston (1978) define lexical simplification as 'the process and/or results of making do with *less* word.' They suggest six strategies that originate from one's semantic competence in their mother tongue. Some strategies use familiar synonyms and paraphrasing when there are cultural gaps between the source and target languages. al Shabab (1996) shows translations have lower lexical variety (i.e., type-token ratio) than originals. Laviosa (1998) studies the narrative translated texts from various source languages into English. It shows translated texts have a relatively lower lexical density (i.e., a lower percentage of content words). Moreover, translated texts use more high-frequency words than originals.

The normalization hypothesis refers to grammatically standardized translation. Toury (1995) suggests a *law of growing standardization* as a main translational behaviour. He suggests that special textual relations in the source texts are replaced by conventional relations in the target texts. One characteristic is avoiding repetition in the originals. Ben-Ari (1998) notes this translation behavior in several ways - cancelling altogether, partially replacing, announcing, or using the variation of the repetitive word in the originals.

The explicitation hypothesis refers to how the implicity in the source texts is made explicit in the translation. Blum-kulka and House (1996) point out that in translations from Hebrew to English, the shifts of cohesive markers by inserting additional words in the target texts raise the level of explicitness. Øverås (1998) studies both translation directions of English and Norwegian and confirms that explicating shifts exist in translations. Koppel and Ordan (2011) show that cohesive markers (e.g., *therefore, thus, hence*) are more frequent in translations than in originals.

Lastly, interference points out the fingerprints of source language usage in the translation. Toury (1979) suggests that translation exhibits the phenomena of *interlanguage* Selinker (1972), which is a linguistic system between the source language and target language, reflecting the interference of these two codes. Such interlanguage is a translation universal presented in the translated texts, and it is even preferred to pure target form with socio-cultural adaption in some situations. Further, Toury (1995) suggests that there is a *law of interference* - 'in translation, phenomena pertaining to the makeup of the source text tend to be transferred to the target text.'

However, the translationese studies above focus mainly on the comparative study between translated and original texts. While translation has different variants, as mentioned in section 2.1, each of them also manifests different translationese phenomena. *Post-editese*, a variant of translationese, represents the distinct characteristics that PE texts exhibit when compared with HT. Due to MT's nature, it tends to select the most frequent words chosen by humans. Farrell (2018) spots MT markers that could be used to distinguish PE from HT in Wikipedia texts translated and post-edited from English into Italian. Although previous studies have shown that human assessors do not seem to distinguish between PE and HT (Daems et al., 2017) and the quality of PE is comparable to HT (Garcia, 2010; Guerberof Arenas, 2009); PE texts show empirical evidence of 'exacerbated translationese.' In terms of simplification, normalization, and interference, PEs are lexically simpler and have more interference from the source language (Toral, 2019).

2.3 CLASSIFICATION BETWEEN DIFFERENT TYPES OF TRANSLATION

2.3.1 Originals vs. human translation

Although the binary classification between original and human-translated texts are not the focus of our experiments, the previous findings inspire our experiments on the classification of different translation types.

The first machine learning classification concerning translationese has been implemented by Baroni and Bernardini (2005). Support vector machines (SVMs) are used to detect original and translated Italian based on translationese features with a varying number of n-grams (unigrams, bigrams, and trigrams) and types (wordform, lemma, part-of-speech and mixed). The mixed representation contains the inflected function words and the part-of-speech tags of the content words. Unigram wordform and lemma representation capture the lexical usage while the mixed representation and wordforms of multiword represent the grammatical information. The best model achieves a high accuracy of 86.7%, and it performs better than professional translators at the same task. Furthermore, it heavily relies on the distribution of n-grams of function words and morpho-syntactic features such as non-clitic personal pronouns and adverbs. This binary classification task verifies the presence of translationese.

Following this line, several studies also implement automatic text classification methods with different translationese features. Ilisei et al. (2010) employ several supervised machine learning classifiers to distinguish between HT and original texts. The classifier is trained with 21 simplification universal features such as average sentence length, type-token ratio, and information load as lexical density. Data in medical and technical domains consists of comparable pairs of translated texts by students, professionals, and original texts. The SVM classifier achieves a high accuracy of 97.62% in distinguishing translated medical texts by professionals and original medical texts, which provides evidence of the existence of simplification features.

Moreover, it also reveals that lexical richness, sentence length, and proportion of grammatical words[1] (i.e., lexical density) are among the most useful features regardless of any classifiers. Some morphological attributes like nouns and pronouns also contribute useful information for classification. Specifically, in translated texts, type-token ratio, mean sentence length, and lexical density are all smaller.

Ilisei and Inkpen (2011) implement several supervised machine learning classifiers including SVM, Naive Bayes, Decision Trees for the binary classification of human-translated and original Romanian texts in the news domain. Besides the features used in 2010, they also add another set of features modelling at the morphological levels in which the translationese effect is claimed to take place (Laviosa, 2002). For example, the proportion of content words (nouns, verbs, adjectives, and

---

[1] They are defined as determiners, prepositions, auxiliary verbs, pronouns, and interjections.

adverbs), pronouns and prepositions are included in this task. Overall, the classifiers have reached considerably good results, ranging from 91.71% to 98.90%. The result shows that information load (i.e., lexical density) is the most useful feature, following the proportion of nouns, proportion of prepositions, and lexical richness.

Koppel and Ordan (2011) use the frequencies of function words taken from LIWC (Pennebaker et al., 2001) as features and demonstrate the phenomenon of explicitation with the prevalence of cohesive markers/adverbs in translated English. Bayesian logistic regression is used for this binary classification task with ten-fold cross-validation, achieving an accuracy of 96.7%. Two categories show a significant difference between original and translations - animate pronouns (e.g., I, we, you, she) and cohesive markers (e.g., therefore, thus, consequently). It suggests that the under-representation of pronouns and the over-representation of *the* in translations might be the result of *explicitation* and *simplification*.

A systematic investigation of translationese features with text classification is employed by Volansky et al. (2013). The data set is EUROPARL (Koehn, 2005) containing 4 million English (original language) tokens and the same number of tokens translated from ten source languages. SVMs with sequential minimal optimization (SMO) is employed as the classification algorithm trained with ten-fold cross-validation. Four translationese hypothesis - simplification, interference, explicitation, and normalization - are modelled through different features. The classification is conducted with each feature separately. They conclude that under four categories of translationese features, interference is the best indicator for translation, with the classification accuracy achieving over 90% based on part-of-speech ngrams. According to the simplification hypothesis, mean word rank achieves the best accuracy of 77%. Features belonging to the explicitation hypothesis do not exceed the classification accuracy of 58%. However, following Koppel and Ordan (2011); Blum-kulka and House (1996), they use 40 cohesive markers (e.g., *moreover, thus, and besides*) as explicitation features, and the classifier achieves an accuracy of 81%. The finding suggests that such cohesive markers are more frequent in translation than in the original texts. Features of normalization such as repetition and contractions perform a bit higher than chance levels because of the scarcity of such features in EUROPARL. However, threshold pointwise mutual information (PMI) capturing the number of associated bigrams whose PMI is above 0 gets better accuracy of 66%. The results show that original English uses more fixed expressions than in translated English.

2.3.2 Machine translation vs. Human translation

As for the classification task of MT and HT, a considerable amount of research has been conducted. However, the vast majority of this research does not take translationese fully into account.

Carter and Inkpen (2012) extract lexical simplification features of type-token ratio, average unigram length, and unigram frequencies. SVMs is used to classify text as human-written English, human-written French, machine-translated English, or machine-translated French. The results show that MT's traits are indeed detectable by machine learning classifiers with 99.8% and 98% accuracy for Canadian Hansard[2] and several Canadian government web sites. They suggest different machine translation systems may produce different sub-optimal translations, and thus building separate models for different MT systems might be needed.

Arase and Zhou (2013) focus on the *phrase salad* phenomenon (Lopez, 2008) of MT. *Phrase salad* means a phrase is fluent and grammatically correct, but when looking into inter-phrases, the correctness of grammar and fluency is poor. This characteristic is captured through fluency, grammaticality, and completeness of the sentences. Fluency features are computed via language models trained on human-written sentences and machine-translated sentences. Grammaticality is calculated using language models trained with the part-of-speech sequences of human-written and machine-translated sentences. Besides, the completeness of non-contiguous phrases, referring as *gappy-phrase* (Bansal et al., 2011), is also modelled as part of the phrase salad phenomenon. The dataset is created by crawling eight websites with human-generated Japanese and English parallel texts translated by Bing Translator, Google Translate, and an in-house SMT system. An SVM classifier is used with a radical basis function kernel because of a smaller number of features. Combining all features, the model achieves an accuracy of 95.8%. It also investigates the generality of those features by applying the same method to the English dataset. They suggest that since English is a morphologically poor language, the effect of features computed by part-of-speech language models may be constrained.

In terms of fluency, Chae and Nenkova (2009) conduct fluency prediction between (fluent) HT and (less fluent) MT based on sentence-level syntactic phrasing features such as parse tree depth, sentence length, and (unnormalized) phrase lengths. There is only one translation direction, Chinese to English, involved in this

---

[2] `https://www.isi.edu/natural-language/download/hansard/`

experiment. The data contains human and machine translations with five levels of human-evaluated fluency scores. Four different classifiers - decision tree, logistic regression, support vector machines, and multi-layer perceptron - are used with 10-fold cross-validation. The result shows that surface structural features provide useful information. In particular, support vector machines perform better when distinguishing worse MT from HT than better MT, achieving an accuracy of 0.86.

Moreover, the classification performance gets better as the divergence of fluency quality increases. Aharoni et al. (2014) further validate the inverse relationship between the quality of the MT systems and the detection accuracy, using features such as part-of-speech n-grams and frequencies of function words taken from (Pennebaker et al., 2001). An SVM classifier is built with SMO as the learning algorithm. To detect the machine-translated texts from different MT systems, they construct the dataset with a portion of Canadian Hansard corpus[3] and translate the French part using Google Translate, Systran, and five other commercial MT systems.[4]. The result shows a strong correlation between the accuracy and the BLEU score or the human evaluation score of the machine translation.

Nguyen-Son et al. (2019) propose a method to distinguish human translation from machine translation based on the coherence score. They assume that human-translated texts are more coherent than machine-translated texts and thus are easier to comprehend. To estimate the coherence, they match similar words in paragraphs with maximum similarity measured by Euclidean distance. The similarity is used as the coherence features to determine whether a human or machine generates a text. The data contains 2000 aligned pairs from TED talks of original English and translated English (from German). An SVM optimized by SMO is used, and it achieves the best accuracy of 87%. They also apply the same approach to Dutch and Japanese, and it shows this method is robust throughout different resource levels.

2.3.3 Human translation vs. Post-edit

Daems et al. (2017) find fully post-edited texts are indistinguishable from HT concerning quality, reader perception, syntactic and semantic translationese features. To construct the dataset, professional and student translators translated eight dif-

---

[3] https://www.isi.edu/natural-language/download/hansard/

[4] http://itranslate4.eu

ferent newspaper articles of roughly 150-160 words long from English into Dutch. The corpus consisted of 87 human translations and 87 post-edited texts. A reader perception study was conducted by asking translation students to 'mark the texts you think are PE'. The results show that humans cannot distinguish between HT and PE. They also implement a computational approach with 55 distinct syntactic and semantic features, including average word length, perplexity, and the average number of content words. The classification results from a logistic regression show that the computer cannot accurately distinguish between HT and PE. However, we consider that this classification result is due to the little amount of training data, preventing the machine from acquiring enough information.

However, Toral (2019) corroborates the presence of post-editese. Three datasets are used in the experiment, Taraxü (Avramidis et al., 2014), IWSLT (Bentivogli et al., 2016; Cettolo et al., 2016) and Microsoft "Human Parity" (Hassan et al., 2018), covering five different translation directions. Four features - lexical variety, lexical density, length ratio, and part-of-speech sequences - addressing different translation universals are computed. Computational analyses prove that PE texts show lower lexical variety/density than HT. This links to the simplification hypothesis that post-editese is lexically simpler than translationese. Sentence length in PEs is more similar to that of the source texts, which may be because of interference and normalization. Moreover, part-of-speech sequences in PEs are more close to the source languages' PoS sequences than HT. This suggests that the interference from source language is more prevalent in PE.

3 DATA COLLECTION

This chapter gives an overview of the data sets used for the classification experiments and language model building. Three types of translations (i.e., HT, MT, and PE) are used to make a cross-comparison and analysis. They are used for the following three classification experiments:

- Distinguishing HT and MT
- Discriminating between HT and PE
- Distinguishing PE and MT

Currently, there are no available data sets that include all the forms we need. So we make use of the following four data sets for different classification tasks: WMT19-submitted-data (Barrault et al., 2019), Microsoft "Human Parity" (Hassan et al., 2018), Automatic Post-Editing (APE) shared task (Chatterjee et al., 2019), and APE-QUEST (Ive et al., 2020).

We would also like to investigate whether having more linguistic information from the neighboring sentences helps distinguish between different translation types. Thus, instead of classifying at sentence-level, sentences are grouped sequentially into text chunks of 2, 5, and 10 sentences as the classification blocks. An overview of the dataset is shown in Table 1.

Section 3.1 introduces the data set from the Conference on Machine Translation (WMT) 2019 news translation task, used for the classification of MT and HT. To inspect the effect of quality mismatch, additional training and testing data generated by DeepL is also used in our experiment. Section 3.2 gives an overview of the data used to classify HT and PE, which is Microsoft 'Human Parity.' Section 3.3 presents the data used in MT and PE classification from two different data sets - APE and APE-quest. Besides the classification task data, we also employ News Crawl 2018/2019 to build the language models for perplexity calculation. Details are presented in section 3.4.

Task	Dataset	Translation Direction	Domain
HT vs. MT	newstest2016-2019	de, fi, gu, kk, lt, ru, zh→en en→de, fi, ru, gu	News
HT vs. PE	MS Human Parity	zh→en	News
PE vs. MT	APE APE-QUEST	en→de, ru en→fr, nl, pt	IT Legal

Table 1: Overview of the data sets used for classification tasks.

3.1 MACHINE TRANSLATION AND HUMAN TRANSLATION DATA

We use publicly available[1] data from the news translation task in Conference on Machine Translation (WMT) 2019. In total, there are eleven translation directions including seven languages (German (de), Finnish (fi), Gujarati (gu), Kazakh (kk), Lithuanian (lt), Russian (ru), Chinese (zh)) translated into English and four languages (de, fi, ru, and gu) translated from English.

The MT's quality is measured through *direct assessments* (DA) by human assessors. They are asked to rate a given MT by 'how adequately it expresses the meaning of the corresponding reference translation or source language input' (Barrault et al., 2019) on a scale of 0 to 100. According to this measurement, an MT system, Facebook FAIR for translation direction of en→de has achieved the highest DA score, 90.3, and it is claimed to achieve super-human performance, while the MT system for en→gu has the lowest DA score of 66.8.

For the training set, the outputs of the best MT systems from previous years (*newstest2016*, *newstest2017* and *newstest2018*) are collected. Since Neural Machine Translation (NMT) outputs are more fluent and closer to the reordering in the reference (Toral and Sánchez-Cartagena, 2017), only the outputs from NMT systems are used in this experiment. For the testing set, we also selected the output of the best MT systems in WMT19 (*newstest2019*). Here we define best systems according to DA score with standardization (Barrault et al., 2019, section 3.7). If more than one system belongs to one corresponding cluster (that is, they do not have a significant difference in DA scores), we select the one with the highest BLEU score (Toral and Sánchez-Cartagena, 2017). Previous studies (Zhang and Toral, 2019; Läubli et al., 2018) point out that the inclusion of translationese in the test sets have inflated DA and BLEU scores compared with the test sets written in the original language. This effect of translationese in the source language might further influence our classi-

---

[1] `http://www.statmt.org/wmt19/metrics-task.html`

fication results. We thus exclude the translationese part by only collecting documents labeled with `origlang="source language"` in the SGM files. SGM stands for Standard Generalized Markup language, a programming language for generating digital documents with custom tags. Moreover, those custom labels provide information about the file's structure and an overview of the content.

Newstest2019 has already excluded the translationese part (Barrault et al., 2019), so the whole data set is used. For language pairs which have data from previous years, we concatenate all the non-translationese parts from previous years together as the training data and use *newstest2019* as the test set. However, for language pairs such as lt→en, kk→en, gu→en and en→gu, the only available data from the year 2019 is split into training (70%), and testing sets (30%) sequentially.

The overview of the best MT systems and corresponding sentence pairs are shown in Table 2. Each sent pair represents a tuple of (*MT, HT, source*).

Direction	**newstest2016 # Sent pairs**	**newstest2017 # Sent pairs**	**newstest2018 # Sent pairs**	**newstest2019 # Sent pairs**	**Train pairs**	**Test pairs**
de → en	UEDIN-NMT 1,499	online-B 1,502	RWTH 1,498	Facebook_FAIR† 2,000	4,449	2,000
fi → en	-	online-B 1,500	NICT 1,500	MSRA.NAO 1,996	3,000	1,996
ru → en	AMU-UEDIN 1,498	online-B 1,499	Alibaba 1,500	Facebook_FAIR 2,000	4,497	2,000
zh → en	-	SogouKnowingnmt 1,000	NiuTrans 2,481	Baidusystem 2,000	3,481	2,000
lt → en	-	-	-	GTCOMPrimary 1,000	700	300
kk → en	-	-	-	NEU 1,000	700	300
gu → en	-	-	-	NEU 1,016	710	306
en → de	UEDIN-NMT 1,500	LMU-nmt-reranked 1,502	FACEBOOK_FAIR 1,500	Facebook_FAIR‡ 1,997	4,502	1,997
en → fi	-	onlineB 1,502	NICT 1,500	GTCOMPrimary 1,997	3,002	1,997
en → lt	-	-	-	tilde-nc-nmt 998	698	300
en → ru	-	onlineB 1,502	Alibaba-ensemble-model 1,500	Facebook_FAIR† 1,997	3,002	1,997

Table 2: The best NMT systems and collected sentence numbers for each language pair. The outputs from the best NMT systems listed above are used in the experiments. The cells with "-" mean this translation direction is not provided in that year. MT systems with "†" have achieved human parity, and the system with "‡" has attained super-human performance.

3.1.1 Machine translation data by DeepL

The previous experiments use data from previous years (*newstest2016*, *newstest2017* and *newstest2018*) as the training set, and thus the translation quality of training data is not as good as *newstest2019*. Such quality mismatch might affect classification accuracy. To eliminate this discrepancy, a fixed MT system, DeepL[2], is used to generate the training and testing data for three language pairs: de→en, en→de and en→ru. DeepL is an NMT-based online translation service trained with the Linguee database to produce more natural word sequences. According to the professional translators, the outputs from DeepL are rated better than other online translation services such as Google Translate, Microsoft Bing, and Amazon Translate.[3]

DeepL translates the source texts from *newstest2016*, *newstest2017* and *newstest2018* as training sets and *newstest2019* as testing sets. Table 3 shows an overview of the data distribution and total training and testing pairs.

Direction	**newstest2016** **# Sent pairs**	**newstest2017** **# Sent pairs**	**newstest2018** **# Sent pairs**	**newstest2019** **# Sent pairs**	**Train pairs**	**Test pairs**
de → en	1,499	1,502	1,498	2,000	4,499	2,000
en → de	1,500	1,502	1,500	1,997	4,502	1,997
en → ru	-	1,502	1,500	1,997	3,002	1,997

Table 3: Data sets statistics for classification between MT generated by DeepL and HT.

3.2 HUMAN TRANSLATION AND POST-EDITS DATA

The Microsoft Human Parity dataset is used for the classification of HT and PE. It contains only one language pair, zh→en. The source texts are from *newstest2017*. Professional translators construct human translation from scratch without using any online translation engines. The post-edits are also annotated by professional translators based on Google Translate machine translation results (Hassan et al., 2018). There are 2,001 pairs of (*HT, PE, source*)[4]. However, only 1,000 pairs translated from the texts originally written in Chinese are used in order to avoid translationese (Toral et al., 2018a).

---

[2] `https://www.deepl.com/translator`

[3] `https://www.deepl.com/press.html`, section "How do we compare to the competition?". (accessed 10-11-2020)

[4] This dataset provides outputs of various NMT systems. However, the MT output used for PE is not provided. Thus, we cannot conduct MT v.s. PE with this dataset.

We would also like to investigate how features contribute differently across different tasks while the translations are from the same source. To examine this, we compare this task with the classification of MT vs. HT. We use the same HT in both settings while the MT outputs are generated by the Microsoft NMT system, *combo-6*. An overview of the dataset is shown in Table 4.

Task	**Train pairs**	**Test pairs**	**MT system**
HT vs. PE	700	300	Google Translate
MT vs. HT	700	300	Combo-6

Table 4: An overview of the datasets used in the both tasks. All translations are derived from the same source *newstest2017* in Chinese but machine-based translations are generated in different ways.

3.3 MACHINE TRANSLATION AND POST-EDITS DATA

We use two publicly available data sets, Automatic Post-Editing (APE) shared task (Chatterjee et al., 2019) and APE-QUEST (Ive et al., 2020). Both datasets provide triplets of (*MT, PE, source*).

The APE dataset is provided by the EU project QT21[5] contains two translation directions (en→de and en→ru) and is from the Information Technology domain. For en→de, MT outputs are produced using the attentional encoder-decoder architecture (Bahdanau et al., 2016), and there are 13,442 training pairs and 1,000 test pairs. For en→ru, MT outputs are generated by the Microsoft Translator production system, and there are 15,089 and 1000 training and testing pairs, respectively. The human post-edits are manually-revised and annotated by professional translators in both translation directions.

There are two common post-editing levels which are differentiated in terms of expected quality - *light post-editing* and *full post-editing* (TAUS, 2016). Light post-editing refers to only major grammatical errors made by MT. They are corrected to make the text comprehensible for the users. It usually has a "good enough" quality that leaves readers aware that it is produced by MT systems. On the other hand, full post-editing aims for human-quality outputs that are stylistically fine and have a "publishable quality" (Hu and Cadwell, 2016; TAUS, 2016). However, in Chatterjee et al. (2019), it is not indicated whether the type of post-editing is light or full.

---

[5] `http://www.qt21.eu/`

Another dataset, APE-QUEST (Ive et al., 2020), provides three translation directions translating English into French (fr), Dutch (nl) and Portuguese (pt) and contains 9,989, 11,249, and 10,165 sentence pairs respectively. The data domain focuses on the areas of online dispute resolution (ODR), procurement, and justice. An NMT system produces MT outputs. Based on these MTs, PEs are annotated by professional translators. Annotators are required to conduct light post-editing. That is, they are only allowed to correct actual grammatical errors rather than improving the writing style. The Translation Edit Rate (TER) (Snover et al., 2006) for this dataset is rather low (i.e., less post-editing is needed) due to the high MT quality of the used NMT system. TER measures the amount of editing that a human would have to perform to change a system output, so it exactly matches a reference translation. It is defined as the minimum number of edits needed to change a hypothesis to exactly match one of the references, normalized by the average length of the references as shown in equation 1. Table 5 presents an overview of the data used for this classification task.

$$\textit{Translation edit rate} = \frac{\textit{\# of edits}}{\textit{average \# of reference words}} \tag{1}$$

Direction	en → de	en → ru	en → fr	en → nl	en → pt
Train pairs	13,442	15,089	6,990	7,870	7,115
Test pairs	1,000	1,000	2,999	3,379	3,050
Domain	IT	IT	Legal	Legal	Legal

Table 5: Dataset statistics for the classification between MT and PE.

3.4 DATA FOR TRAINING LANGUAGE MODELS

Two language models are built to calculate perplexity - one for surface form and another for the part-of-speech sequences. Since an English trigram language model could be used to compute the perplexity for almost half of the translation directions in the experiments[6], we only built the language model for perplexity in English. We decided to use News Crawl (Barrault et al., 2019) since the language pairs translated into English in our experiments are in the news domain. News Crawl 2018 and

---

[6] We have eleven total translation directions in the task of MT/HT, one total translation direction in the task of HT/PE, and five total translation directions in MT/PE. There is eight translation directions' target language is English.

2019 in English are first concatenated together, resulting in a total of 51,714,108 sentences, in which 40,000,000 sentences are randomly selected to build a trigram language model using KenLM (Heafield, 2011). For more details about language model building, please refer to section 4.1.1.

For the language model of part-of-speech sequences, 2,000,000 sentences are randomly selected from News Crawl 2019 in five languages - en, de, lt, ru, and zh. We then conduct part-of-speech tagging using UDpipe (Straka, 2018) to get the part-of-speech tags for each word in the sentences. Subsequently, the sequences of part-of-speech are used to build tri-gram language models for each language. Table 6 shows an overview of the training data used for building language models.

Language model	**Data source**	**# of sent**
Perplexity	News Crawl 2018 & 2019	40,000,000
PoS perplexity	News Crawl 2019	2,000,000

Table 6: Data statistics of the training data for the language models. Final training sentences are randomly selected from News Crawl.

4 EXPERIMENTS

This chapter introduces the definition of each translationese feature and the structure of the classification pipeline. In section 4.1, features are categorized and presented according to their associated translation universals. Additionally, how they are calculated and the hypothesis we hold are presented. Section 4.2 explains the whole classification pipeline, including the implementation of Feature Union, Grid Search, Recursive Feature Elimination (RFE), and the classification algorithm - Support vector machines (SVMs).

4.1 TRANSLATIONESE FEATURES

Since our goal is to find which aspect of the translationese has the most distinctive and machine-learnable characteristics between different translations, we follow the framework of Volansky et al. (2013) in which features are categorized according to four translationese universals - simplification, explicitation, interference, and normalization. Although in Volansky et al. (2013), those features are used for the classification of originals and HT, we would also like to explore whether features representing different translation universals are still effective for distinguishing different types of translation than just originals and HT.

In addition to the features mentioned above, we also implement other translationese features. Some of them are found distinctive when comparing HT with PE (Toral, 2019), or used to model the fluency of MT (Toral and Sánchez-Cartagena, 2017), represented as a more robust measure of lexical variety (Vanmassenhove et al., 2019), and used to illustrate the syntactic relations (Kunilovskaya and Kutuzov, 2017). Table 7 shows an overview of all 28 features and their associated hypothesis and the hypothetical values of features in each translation type. For a complete feature list per language pair, please see Appendix A.

4.1.1 Simplification

The simplification hypothesis refers to the phenomenon that translated texts are lexically and syntactically simpler. In total, 13 features are modeling the lexical or syntactic characteristics under the simplification hypothesis.

LEXICAL VARIETY As previous studies (Vanmassenhove et al., 2019; Toral, 2019) point out, MT/PE have a lower degree of lexical richness than HT. We use three metrics to evaluate lexical richness: type-token ratio (TTR), Yule's I (the reverse of Yule's K) (Yule, 1944), and the measure of textual lexical diversity (MTLD) (McCarthy, 2005). TTR is the ratio between unique words and total words as shown in equation 2.

$$TTR = \frac{\textit{number of unique words}}{\textit{number of total words}} \tag{2}$$

While TTR is only comparable between similar size texts due to its assumption of the linear relation between types and tokens, Yule's K and its reverse Yule's I are less sensitive to the varying text length (Oakes and Ji, 2012). It is calculated as shown in equation 3 where V denotes the number of unique tokens and $f_v(i,N)$ refers to the numbers of types occurring i times in a sample of length N (Benoit et al., 2018).

$$\begin{aligned} \textit{Yule's } I &= \frac{V^2}{M_2 - V} \\ M_2 &= \sum_{i=1}^{V} i^2 \times f_v(i,N) \end{aligned} \tag{3}$$

Another metric, MLTD, is defined as 'the mean length of sequential word strings in a text that maintains a given TTR value.' It is proved to be a powerful and robust indicator regarding text length (Mccarthy and Jarvis, 2010). A higher value of TTR/Yule's I/MTLD indicates a richer vocabulary, and HT exhibits this tendency when compared with MT/PE. Based on these findings, we expect to see higher values of those three metrics for HT when compared with MT and PE. Moreover, PE will have a more diverse vocabulary than MT.

AVERAGE WORD LENGTH We calculate the ratio between the counts of characters and sentence tokens, representing the average word length in a sentence, as shown in equation 4. We hypothesize that MT uses simpler (shorter) words than HT and PE.

$$\textit{Average word length} = \frac{\textit{number of total characters}}{\textit{number of total words}} \tag{4}$$

SYLLABLE RATIO This measure is a variant of the average word length. The hypothesis is that translated texts use simpler words that contain fewer syllables per word. We expect this phenomenon is more obvious in MT and PE than in HT. A Python module called Big Phoney[1] is used to calculate the counts of syllables per word. The syllable-counting accuracy could reach 100% for 134,000 words in The CMU Pronouncing Dictionary[2] and 98.1% for words not in the dictionary. As Big Phoney only supports English, this feature is only available for translation directions into English. Since syllables per sentence are the sum of each word's syllables in the sentence, we sum up the counts of syllables per sentence in each chunk and normalize the value by the total number of sentences in the chuck.

$$\textit{Syllable ratio} = \frac{\textit{sum of syllables per sentence in the chunk}}{\textit{number of total sentences}} \tag{5}$$

AVERAGE SENTENCE LENGTH In Ilisei et al. (2010), the average sentence length is among the most informative of the simplification features. We hypothesize that MT/PE will have a shorter sentence length than HT. That is, they contain fewer tokens per sentence as they are supposed to be syntactically simpler. Additionally, MT will have a shorter sentence length than PE. We calculate this feature in token levels, as shown in equation 6.

$$\textit{Average sentence length} = \frac{\textit{number of total tokens}}{\textit{number of total sentences}} \tag{6}$$

LEXICAL DENSITY Lexical density reveals how much information is presented in each text chunk. It is shown to be lower in HT than in originals (Laviosa, 1998) and lower in MT/PE than in HT (Toral, 2019). It is also an informative feature (Ilisei et al., 2010) when classifying between originals and HT. The hypothesis is

---

[1] `https://pypi.org/project/big-phoney/`

[2] `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

that MT/PE will have a lower lexical density than HT; moreover, MT will have an even lower lexical density than PE.

It is calculated as the number of content words (adverbs, adjectives, nouns, and verbs), normalized by the total number of tokens in the chunk, as shown in equation 7. To label content words, each word is tagged with its Universal part-of-speech tags via SpaCy+UDPipe[3].

$$\textit{Lexical density} = \frac{\textit{number of content words}}{\textit{number of total words}} \tag{7}$$

AVERAGE WORD RANK Unlike HT, MT and PE tend to choose the most frequently occurring words in the training data repeatedly (Farrell, 2018). We hypothesize that MT/PE will use more frequent words than HT. That is, MT/PE will have a lower mean rank than HT. Furthermore, MT will have an even lower rank than PE.

Following Volansky et al. (2013), we extract the top 5,000 frequently used words in seven languages[4] via a Python library called *wordfreq* (Speer et al., 2018). Each word's rank is summed up for each sentence, normalized by the number of total words. Punctuation is removed, and if a word does not appear in the ranking list, a rank of 6,000 is assigned.

$$\textit{Average Word Rank} = \frac{\textit{sum of word ranks}}{\textit{number of total words}} \tag{8}$$

MOST FREQUENT WORDS This feature is a variant of the average word rank. We hypothesize that among the three types of translation (MT, PE, and HT), frequent words (e.g., *the* and *of* in English) are used more often in MT than PE, while in HT, those words are less common. This leads to a higher value of this feature in MT, followed by PE and HT.

We render this property by using the same Python library *wordfreq* to extract lists of the top 5, 10, and 50 most frequently used words in seven languages. The total number of words appearing in the list are normalized by the total number of words

---

[3] `https://github.com/TakeLab/spacy-udpipe`

[4] It includes English, German, Russian, French, Dutch, Portuguese, and Finnish.

in the text chuck. Punctuation is removed, and the value is calculated separately for three different thresholds of frequent words.

$$\textit{Most frequent words} = \frac{\textit{counts of words appearing in the word frequency list}}{\textit{number of total words in the chunk}} \tag{9}$$

MEAN DEPENDENCY DISTANCE (MDD) Dependency distance is defined as 'the distance between words and their parents, measured in terms of intervening words' (Hudson, 1995). MDD is further proposed as a metric of linguistic complexity, calculated as shown in equation 10 (Liu, 2008).

$$\textit{MDD(the sentence)} = \frac{1}{n-1}\sum_{i=1}^{n}|\mathrm{DD}_i| \tag{10}$$

Here n denotes the numbers of tokens in a sentence. DD_i denotes the dependency distance of i-th token. For *root*, DD_i is zero. We hypothesize that machine translations will use simpler syntactic structures, resulting in a lower degree of MDD.

PERPLEXITY Inspired by Toral and Sánchez-Cartagena (2017), we use perplexity as a proxy for fluency. It is also used in Čulo and Nitzke (2016) as a measure to identify the level of variation of the terminology used in MT, PE, and HT.

As shown in equation 11, it is calculated as the probability of the given texts, normalized by the number of words. For a given sequence $W = W_1W_2...W_N$, its perplexity is calculated as follows:(Jurafsky and Martin, 2009, section 4.4)

$$\textit{PP(W)} = P(W_1W_2...W_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(W_1W_2...W_N)}} \tag{11}$$

We use *KenLM* (Heafield, 2011) to build the tri-gram language models trained on 40 million randomly selected sentences from *News Crawl 2018* and *News Crawl 2019* in English. KenLM is a fast and memory-efficient implementation for building n-gram language models with Kneaser-Ney smoothing. Since perplexity is the inverse of the likelihood of the given sequence, lower perplexity represents more fluent and native language usage. We hypothesize that MT will have higher perplexity (i.e., less fluent), followed by PE and HT.

4.1.2 Explicitation

HT has shown to be more explicit than the original texts, for example, by means of cohesive markers like *therefore* and *thus* (Koppel and Ordan, 2011) to establish an explicit relationship across sentences. In this section, we list four features that can be used to instantiate the phenomenon of explicitation.

EXPLICIT NAMING One explicitation phenomenon is that pronouns are substituted with more explicit proper nouns (Olohan and Baker, 2000; Volansky et al., 2013). Following Volansky et al. (2013), the explicit naming is calculated as the ratio between proper nouns and pronouns. Because human translators generally conduct this substitution, we hypothesize that MT will have fewer proper nouns (i.e., less explicit) than PE and HT, resulting in a higher value.

$$\mathit{Explicit\ Naming\ Ratio} = \frac{\mathit{number\ of\ personal\ pronouns}}{\mathit{number\ of\ proper\ nouns}} \tag{12}$$

SINGLE NAMING HT might add explicit information such as the last name or roles to a person's first name to make it easier to comprehend for readers with different cultural backgrounds. Single naming refers to standing alone proper nouns, i.e., the previous and next word's PoS tags are not labelled as *PROPN*. Since adding extra information is mainly performed by human translators, we hypothesize that MT will have more counts of single proper nouns, followed by PE and HT.

$$\mathit{Single\ naming} = \mathit{counts\ of\ standing\ alone\ proper\ nouns} \tag{13}$$

MEAN MULTIPLE NAMING As mentioned above, if we hypothesize that HT presents more information than MT/PE in the texts by presenting a greater number of proper nouns, then HT will have higher values of multiple naming. Multiple naming refers to the use of consecutive proper nouns in a sentence. Numbers of proper nouns that are consecutive are summed up and normalized by the total number of such consecutive units in each text chunk.

$$\mathit{Mean\ multiple\ naming} = \frac{\mathit{number\ of\ n\text{-}grams\ of\ proper\ nouns}}{\mathit{number\ of\ proper\ noun\ blocks}} \quad \text{where} \quad n > 1 \tag{14}$$

FUNCTION WORD RATIO Function words serve as elements modifying the content words they are attached to. In Koppel and Ordan (2011), two categories of function words show a significant difference between originals and HTs - pronouns and cohesive markers. Cohesive markers help connect the different parts of the text more closely together, such as *therefore* and *consequently*. We hypothesize MT uses more function words than HT and PE because it might use more cohesive markers and pronouns.

Function words in our experiment are selected according to Universal Dependencies (UD) relations. Six UD relations are chosen based on their structural categories[5]: *aux, cop, mark, det, clf, case.* We calculate the ratio between total occurrences of function words and the total number of tokens, as shown in this equation:

$$Function\ Word\ Ratio = \frac{number\ of\ function\ words}{number\ of\ total\ words} \tag{15}$$

4.1.3 Normalization

The normalization hypothesis refers to 'the tendency to conform to patterns and practices which are typical of the target language, even to the point of exaggerating them' (Baker and Somers, 1996). We model this phenomenon with the following three features.

REPETITION A study has shown that translations tend to avoid repetition by eliminating them altogether or replacing them with synonyms (Ben-Ari, 1998). This feature is realized by calculating content words appearing more than once and normalized by the number of total tokens. As indicated in Volansky et al. (2013), *be* and *have* are excluded as they usually function as auxiliaries. We hypothesize that MT will have a higher repetition ratio than PE and HT due to the nature of frequent word usage. On the other hand, HT might have a lower repetition ratio than PE.

$$Repetition\ Ratio = \frac{number\ of\ content\ words\ occurring > 1}{number\ of\ total\ words} \tag{16}$$

AVERAGE POINTWISE MUTUAL INFORMATION (PMI) Translations tend to use certain collocations. That is, some fixed phrases are overused. PMI is a measure of

---

[5] `https://universaldependencies.org/u/dep/`

association of two events (words) by considering the probability of each independent event as shown in equation 17 where $P(w_1, w_2)$ refers to the probability of a co-occurrence of w_1 and w_2 and $p(w_1)p(w_2)$ to the product of a single probability of occurrence. The greater the PMI value, the more likely these two words frequently occur together as a collocation.

$$PMI(w_1, w_2) = log \frac{p(w_1, w_2)}{p(w_1)p(w_2)} \tag{17}$$

Following Volansky et al. (2013), we calculate the mean PMI of all bigrams. We hypothesize that MT will use more collocations than PE and HT, thus resulting in a higher average PMI.

THRESHOLD PMI When PMI equals 0, it indicates that these two words co-occur just by chance; that is, they do not form a collocation. To exclude this arbitrariness, we count the numbers of bigrams with PMI above 0, normalized by the number of all bigrams.

$$\text{Threshold PMI} = \frac{\text{number of bigrams whose PMI} > 0}{\text{number of total bigrams}} \tag{18}$$

4.1.4 Interference

Interference refers to the footprints of source languages in the translations. The source language 'shining through' (Teich, 2003) is modelled by the following six features with regard to part-of-speech sequence, sentence length, and function words.

PART-OF-SPEECH (POS) N-GRAMS We hypothesize that MT/PE has more interference from the source language compared with HT. Therefore, MT/PE will have a more similar PoS tag sequence to the source language. To model this, we construct PoS bigram models[6].

The value of this feature is the term frequency–inverse document frequency (tf-idf) weight for each part-of-speech bigram. A higher tf-idf weight means a certain word appears more frequently in the given document but is not common in the other documents in the corpus. This signifies that this word has a higher word

---

[6] We also experiment with unigram and trigram models. However, it does not help with the classification, so we only use bigrams.

importance and is more relevant for the given document. The formula of tf-idf is as follows, where *i* refers to word and *j* to document:

$$W_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \tag{19}$$

$$tf_{i,j} = \text{number of occurrences of i in j}$$
$$df_i = \text{number of documents containing i}$$
$$N = \text{total number of documents}$$

We use `TfidfVectorizer` in `scikit-learn` to convert each text chunk into a matrix of hundreds of tf-idf weights of PoS bigrams. However, we do not include this feature in our final feature analysis in section 5.2 of this thesis. It might bias our conclusion because this 'Bag of bigrams' representation is not comparable with the other translationese features computed as one final value per text chunk.

CHARACTER N-GRAM Apart from part-of-speech bigrams, we also compute tf-idf weights based on character n-grams that capture the given texts' morphological characteristics. The counts of unigram, bigram, and trigram in the whole corpus are first calculated, and then the tf-idf weight of each character n-gram is derived. Nevertheless, as stated above, we do not include this feature in the feature analysis.

PERPLEXITY OF POS SEQUENCE Following Toral (2019), we assess the interference by calculating the difference of the perplexities derived from two language models, which are trained on the PoS-tagged corpus of the source and target language, respectively. If the PoS tag sequences of a given text are more similar to the source (i.e., it has lower perplexity regarding the source language), then the value of this feature will decrease, and vice versa. We hypothesize that MT/PE will have lower values than HT since they have more interference from the source languages.

$$\textit{PP difference} = \textit{PP(T,LMsource)} - \textit{PP(T,LMtarget)} \tag{20}$$

LENGTH RATIO Length ratio has been proven to be lower for PE and MT compared with HT (Toral, 2019). The hypothesis is that since HT is translated from scratch, the length of HT might be less restricted by the length of source texts, resulting in a higher value of length ratio. It is calculated as the absolute difference in characters between source texts S and target texts T, normalized by the length of the S. If length ratio is lower, the difference between S and T is also smaller. We hypothesize that this feature will be the main discriminator between HT and MT/PE.

$$Length\ Ratio = \frac{|\,length_S - length_T\,|}{length_S} \tag{21}$$

CONTEXTUAL FUNCTION WORD This feature is seen as a variant of PoS n-grams and has achieved 100% of accuracy in the classification task of HT and original texts in Volansky et al. (2013). In that paper and Koppel and Ordan (2011), a list of function words is used for computing the frequency of function word trigrams. However, to identify function words in a more scalable way, we adopt a slightly different approach using universal dependency relations. We calculate the trigrams' counts where at least two tokens' universal dependency relations are labeled as one of the *'aux,' 'cop', 'mark', 'det', 'clf', 'case'*. The value is then normalized by the total number of trigrams in each text chunk. Our hypothesis is that more trigrams in MT/PE are anchored by specific function words than in HT.

$$Contextual\ function\ words = \frac{number\ of\ trigrams\ consisting\ function\ words}{number\ of\ total\ trigrams} \tag{22}$$

POSITIONAL TOKEN FREQUENCY The motivation for this feature is that human translators tend to be conservative in their word choices (Kenny, 2001) and have limited vocabulary to open or close a sentence.

Firstly, we calculate the frequency for words located at the beginning (first, second) and ending (antepenultimate, penultimate, and last) position in the whole corpus, normalized by the total number of words occurring in those positions. This shows 'how frequently this word is occurring at those certain positions.' For example, if *I* is used more often for opening the sentence, then its normalized frequency at this step will be higher. Secondly, for each sentence, the positional token frequency is the sum of the normalized frequency of each sentence's opening and ending words.

For actual computation, we first derive a frequency list FL of all words T at position 1, 2, n-2, n-1, n in every sentence in the corpus. Then given a sentence

$S = W_1 W_2 ... W_n$, the positional token frequency is calculated as follows, where W_i presents the counts of word W at position i in FL, and N denotes the sum of frequency in FL:

$$\textit{Positional token frequency (sentence)} = \sum_{i=1}^{2} \frac{W_i}{N} + \sum_{i=0}^{2} \frac{W_{n-i}}{N} \tag{23}$$

We hypothesize that MT/PE will have a stronger tendency to use certain words to open or end a sentence, resulting in a higher positional token frequency than HT.

4.1.5 Others

Here we list two features that do not fall into any of the translationese hypotheses, but they instantiate the translationese's traits.

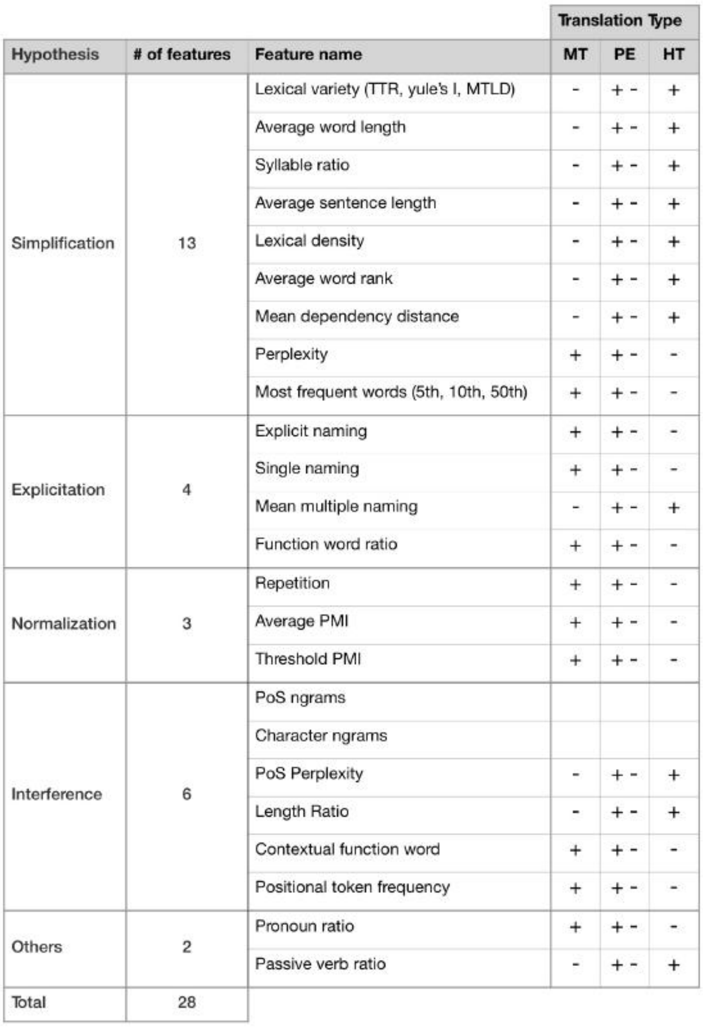
PRONOUN RATIO In Koppel and Ordan (2011), pronoun ratio is among the most distinguishing features between HT and originals. We compute this feature by first identifying the pronouns with PoS tag as *PRON*. Subsequently, for each text chunk, the frequency of pronouns is normalized by the total word counts. Since this feature is somehow related to the previous feature, explicit naming, we hypothesize that MT/PE will have more pronouns than HT since human translators might replace pronouns with other phrases.

$$\textit{Pronoun ratio} = \frac{\textit{number of pronouns}}{\textit{number of total words}} \tag{24}$$

PASSIVE VERB RATIO We defined the structure of passive verbs as either a bigram of *AUX,VERB* or a trigram of *AUX*, *ADV*, *VERB*. We compute this feature by counting the number of such formations, normalized by the total number of verbs. Each sentence is iterated in each text chunk, and the counts of passive verbs are summed up. The final value is normalized by the total number of verbs in each text chunk. As passive voice's goal is to emphasize the importance of the action receiver, we hypothesize that human translators would adopt this translation strategy rather than translating literally. Thus, HT will contain more passive voice than MT/PE.

$$\textit{Passive verb ratio} = \frac{\textit{number of passive verbs}}{\textit{number of total verbs}} \tag{25}$$

To sum up, Table 7 summarizes the indication of feature values that whether greater/smaller values are exhibited in MT/PE/HT.



			Translation Type		
Hypothesis	# of features	Feature name	MT	PE	HT
Simplification	13	Lexical variety (TTR, yule's I, MTLD)	-	+ -	+
		Average word length	-	+ -	+
		Syllable ratio	-	+ -	+
		Average sentence length	-	+ -	+
		Lexical density	-	+ -	+
		Average word rank	-	+ -	+
		Mean dependency distance	-	+ -	+
		Perplexity	+	+ -	-
		Most frequent words (5th, 10th, 50th)	+	+ -	-
Explicitation	4	Explicit naming	+	+ -	-
		Single naming	+	+ -	-
		Mean multiple naming	-	+ -	+
		Function word ratio	+	+ -	-
Normalization	3	Repetition	+	+ -	-
		Average PMI	+	+ -	-
		Threshold PMI	+	+ -	-
Interference	6	PoS ngrams			
		Character ngrams			
		PoS Perplexity	-	+ -	+
		Length Ratio	-	+ -	+
		Contextual function word	+	+ -	-
		Positional token frequency	+	+ -	-
Others	2	Pronoun ratio	+	+ -	-
		Passive verb ratio	-	+ -	+
Total	28				

Table 7: Features and their hypothetical values in different translation types. '-' indicates the lowest value, '+ -' the relatively higher value, and '+' the highest value. PoS and character n-grams are represented as matrices of hundreds of tf-idf weights so the hypothetical values are not applicable here.

4.2 CLASSIFICATION WITH SUPPORT VECTOR MACHINES

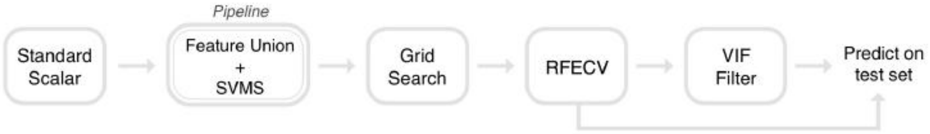


Figure 1: Experiment scheme.

This section presents the experiment structure, as shown in figure 1. Since our goal is to find the most distinguishing features, we construct a machine-learning classifier to identify which features are more distinctive between different translations and explore whether such differences are machine-learnable. Section 4.2.1 shows how feature values are standardized via `StandardScalar` to avoid the classification bias and describes how `Feature Union` works with `Pipeline` to chain the features and classifier together. Section 4.2.2 briefly introduces the definition of the implemented algorithm - Support Vector Machines (SVMs).

Section 4.2.3 presents the hyperparameter tuning process utilizing grid search. A hyperparameter is a parameter that controls the learning process, and it has to be tuned to make the machine-learning algorithm achieve the optimal solution for the classification problem. After the best hyperparameters are found, a feature selection technique used in the experiment, recursive feature elimination (RFE), is presented in section 4.2.4. In the following section 4.2.5, we also investigate the effect of multicollinearity and eliminate features whose Variance Inflation Factor (VIF) are higher than 5 to prevent volatile coefficients, which will further affect the interpretation of feature importance.

To summarize, we conduct the classification using three different subsets of features, as shown below. The results from the first two classifications are the main focus and used for further feature analysis. We exclude PoS and character n-grams from the analyses because the representation of hundreds of tf-idf weights is not comparable with other feature values and it might overshadow the importance of other features.

1. Features (without PoS n-grams and character n-grams) with RFE
2. Features (without PoS n-grams and character n-grams) with RFE and VIF filter
3. All features with RFE

4.2.1 Pipeline and feature union

The features as mentioned earlier in section 4.1 are computed and put into one CSV file so that the training examples have the shape of (*number of samples*, *number of features*). We extract available features from the CSV file and apply `standardscalar`[7] to each of them. The idea of `standardscalar` is to normalize the data before applying the machine learning model to have the mean of 0 and a standard deviation of 1 for each feature. Since our features are calculated at different scales, the standardization of a dataset is necessary. One advantage of normalizing is to eliminate the effect of domination of features with greater numeric ranges. Another advantage is to avoid calculation difficulties as the inner products of big features vectors might cause numerical problems (Hsu et al., 2003). The standard score of a sample `x` is calculated as in equation 26 where u is the mean and s is the standard deviation of the training samples.

$$z = \frac{x - u}{s} \tag{26}$$

After the standardization, `Feature Union`[8] is used to concatenate several feature matrices into a composite feature space. We use the `Feature Union` together with `Pipeline`[9] to pass the large feature matrix to the classifier for training. `Pipeline` chains several estimators (i.e., the base object implements a `fit` method to learn from the data) into a final composite estimator. The last estimator in the pipeline is a SVMs classifier for training the model and predicting on the test set.

4.2.2 Support vector machines (SVMs)

The support vector machine constructs a hyperplane (i.e., dicision boundary) in N-dimensional space that has the maximum margin between the data points of different classes. The margin is defined as the perpendicular distance between the hyperplane and the closest data points. Given training samples $x_1, ..., x_N$, with corresponding target values $t_1, ..., t_N$ where $t_n \in -1, 1$. New data points are classified

---

[7] `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html`

[8] `https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.FeatureUnion.html`

[9] `https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html#sklearn.pipeline.Pipeline`

according to the sign of equation 27 where $\phi(x)$ denotes a fixed feature-space transformation, w^T is the weight vector, and b is the bias (Bishop, 2006).

$$y(x) = (w^T\phi(x) + b) \tag{27}$$

The objective of the support vector machine algorithm is to find the maximum-margin hyperplane that separates training points whose $t_n = 1$ from training points whose $t_n = -1$. By maximizing the margin, the classifier's generalization error will be lower, and thus the new data points can be predicted more correctly. As shown in Figure 2, any hyperplane can be defined as $(w^T\phi(x) + b) = 0$ and the training points on the margin are called support vectors.

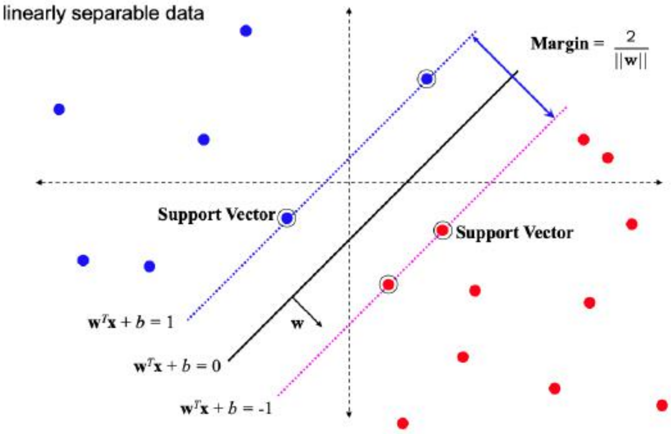

linearly separable data
Margin = $\frac{2}{||\mathbf{w}||}$
Support Vector
Support Vector
$\mathbf{w}^T\mathbf{x} + b = 1$
$\mathbf{w}$
$\mathbf{w}^T\mathbf{x} + b = 0$
$\mathbf{w}^T\mathbf{x} + b = -1$

Figure 2: SVM (Zisserman, 2015)

After some trial and errors, we find non-linear kernels do not perform better than linear kernel. Thus, Linear Support Vector classification (`LinearSVC`) is used in our experiment. `LinearSVC` is similar to `SVC` with linear kernel but faster liblinear[10] implementation for data with millions of instances and features. The input data takes training samples of shape (*number of instances*, # *of features*) and class label of shape (*number of instances*).

4.2.3 Grid search and cross-validation

The estimators' hyperparameters are not directly learned during the training, so they are fine-tuned via an optimization process. The grid search provided by `GridSeachCV` conducts an exhaustive search over all parameters combinations. That

---

[10] https://www.csie.ntu.edu.tw/~cjlin/liblinear/

is, the model is fitted on the dataset with all possible combinations of hyperparameters. The best hyperparameter combination is the one that yields the best cross-validation score on the training set. To make sure the results are reproducible, `random state`, the seed used by the random number generator is set to 42 for all combinations. We define the values of hyper-parameters, as shown in Table 8. We decide to tune five hyperparameters because they affect the calculation of the optimal hyperplane. We experiment with a set of different values decided upon the hyperparameter's nature; for example, a larger *C* denotes a smaller margin and vice versa. Furthermore, *tol* is the tolerance for stopping criterion; that is, when the loss is not improving by at least the threshold, the searching for hyperplane and training will stop.

Hyperparameter	Value
C	2,5,100,1000
loss	hinge, squared_hinge
class_weight	balanced, None
tol	0.001, 0.0001
max_iter	2000, 10000, 100000

Table 8: Values of hyperparameters that are used in the parameter grid for hyperparameter tuning.

All possible combinations of the hyper-parameters are passed and fitted onto the estimator. Cross-validation is used together with grid search in `GridSearchCV` to calculate the best validation score. While training the model with limited data, cross-validation is a strategy to evaluate the model's generality by predicting unseen data. It first splits the dataset equally into *K* folds and use *K-1* folds as training data. The resulting trained model is then evaluated on the remaining (unseen) part of the dataset, as shown in Figure 3.[11] This procedure goes on *K* consecutive times until each *K* fold is evaluated. In `GridSearchCV`, each parameter combination is trained and evaluated *K* times. The score of each combination is obtained by calculating the mean cross-validated score. The best parameter setting is found with the best performing estimator, which gives the highest score on the left-out data. We perform a grid search with 10-fold cross-validation to find the optimal hyper-parameters for the LinearSVC classifier.

---

[11] `https://scikit-learn.org/stable/modules/cross_validationhtml`

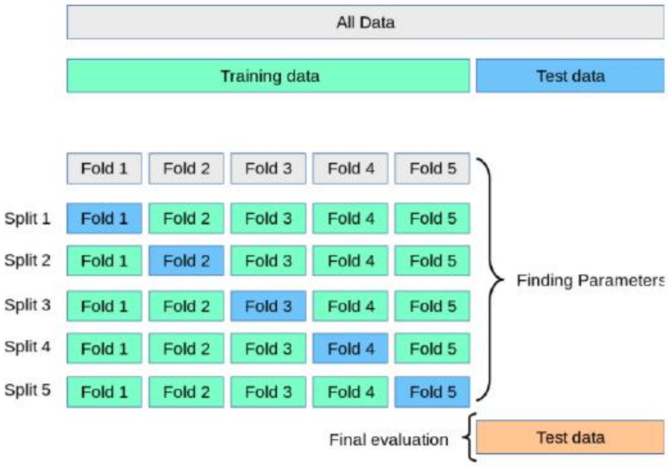


Figure 3: Visualization of cross-validation technique from scikit-learn website. This figure visualizes 5-fold cross-validation while in our experiment we employ 10-fold cross-validation.

4.2.4 Recursive feature elimination

We then use the best hyper-parameters to perform feature selection. Recursive feature elimination (RFE, Guyon et al. (2002)) is a feature selection technique that eliminates the least important features recursively until the desired feature numbers are reached. It begins with training the model with all features, then the features are ranked according to their importance, and the least important feature(s) are eliminated. The model is re-built with the new subset of features, and the feature importance is calculated again. However, the optimal feature numbers is another tuning parameter for RFE (Butcher and Smith, 2020), and it is usually hard to estimate in advance the optimal features numbers. We thus employ `RFECV`[12] which combines RFE with cross-validation to test on different subsets of features and pick a subset of features that yields the best cross-validation score.

The feature importance is learned during the fitting, and it could be accessed in scikit-learn through a `coef_` attribute for a linear SVM. The value given by `coef_` is the weight vector w^T in equation 27 which is orthogonal to the hyperplane. This weight vector gives the direction for predicting classes by taking the dot product of any data points with it. Specifically, if a feature obtained a negative coefficient, it contributes to the classification of the negative label and vice versa. We also take

---

[12] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFECV.html#sklearn.feature_selection.RFECV

the absolute values of coefficients, which indicate how important the features are for the classification.

4.2.5 Variance Inflation Factor (VIF)

If there is a correlation between the predictors (i.e., independent variables) in a regression model, the effect of multicollinearity might be present. Multicollinearity will result in unstable coefficients since they will change dramatically with a small change of variables in the model (Montgomery et al., 2012). There are two types of multicollinearity, one is structural, and another is data-based. Structural multicollinearity refers to the correlation between two variables in which one of them is derived from another. For example, a predictor X^2 will be highly correlated with X. Data-based multicollinearity usually happens with observational data rather than an artifact of the model. Since some of our features in the experiments might be correlated, e.g., Type token ratio and Yule's I are both computing the degrees of the lexical variety but in different ways, we inspect the effect of structural multicollinearity through Variance Inflation Factor (VIF).

The VIF estimates how much the standard errors are inflated. The VIF of an estimated coefficient b_j, denoted as VIF_j, means what percentage of the variance of b_j is inflated by the correlation among all the predictors in the model. The VIF of estimated coefficient b_j is calculated as in equation 28 where R^2 value is calculated by regressing each of the predictors against every other predictor in the model (Dodge, 2008).

$$VIF_j = \frac{1}{1 - R^2} \tag{28}$$

The numerical value of VIF ranges from one, and if there is no multicollinearity, the value of VIF equals one, meaning the variance is not inflated. A rule of thumb for the interpretation of VIF is if it is higher than five, then a further investigation is needed as it suggests high correlation relationships exist. A value between one to five indicates a moderate correlation between the specific predictor and other predictors.

For our experiment, we import a Python package `statsmodels`[13] for the computation of VIF. VIF values are calculated for all features, and we eliminate the feature

---

[13] `https://www.statsmodels.org/stable/index.html`

with the highest VIF. This VIF calculation and feature elimination are repeated recursively until there is no VIF value higher than 5.

5 RESULTS AND DISCUSSION

This chapter reports the results and analysis for three classification tasks (i.e., MT/HT, MT/PE, HT/PE). Section 5.1 reports the classification accuracy using three different subsets of features as listed in section 4.2. In section 5.2, we analyze feature importance in two ways: 1) feature ranking based on absolute coefficients and 2) features' average coefficients, across all translation directions. Feature ranking refers to 'how important a feature is' for the classification task since greater absolute coefficients represent greater weights for prediction. Subsequently, if a feature is found to be useful, then 'which label this feature contributes to classify' is implied by the sign of the coefficients.

5.1 CLASSIFICATION RESULTS

5.1.1 Machine translation vs. human translation

Figure 4 shows the accuracy predicted on the test set using features excluding PoS and character ngrams. The best accuracy of 0.76 is achieved for gu→en with a context length of 10. However, it could be due to overfitting as there are only 62 instances (31 instances for each label) used in the test set. The same trend could be spotted with four other language pairs (gu,kk,lt→en and en→lt, hereafter **LP-2019**) trained and tested on the same dataset of *newstest2019*. They generally perform better than other translation directions trained with previous years (hereafter **LP-prev**).

Figure 5 shows the cross-validation scores during RFE for both groups. For LP-prev, the cross-validation scores are more stable, but the performance on average drops more (-0.2) than LP-2019 (-0.05) when predicting on the unseen data. We expect this might be due to the quality mismatch between training and testing data as mentioned in section 3.1.1. To further investigate this, we generate data via DeepL and report the classification results in section 5.1.2.

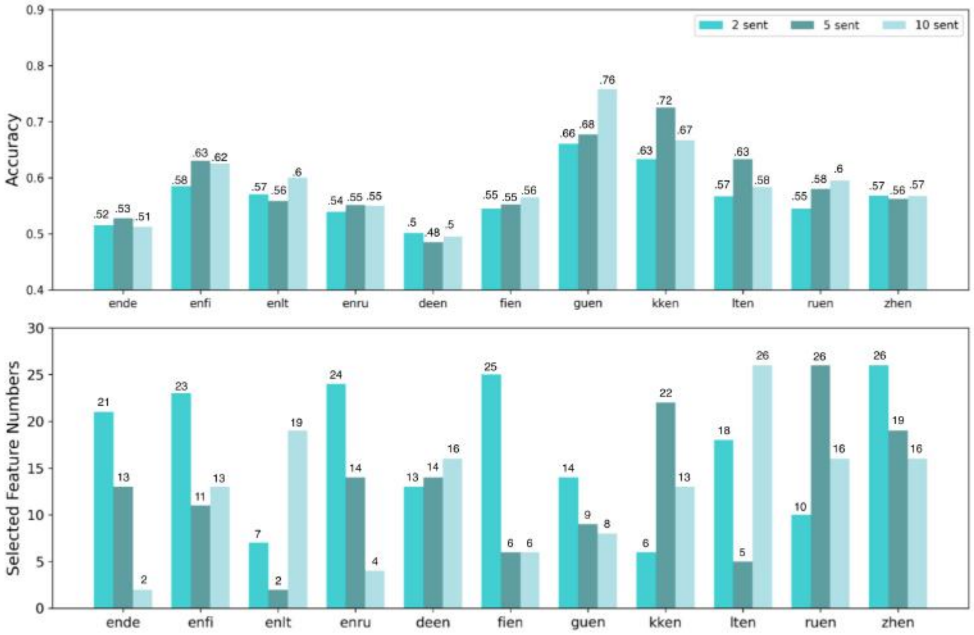


Figure 4: MT-HT classification results on test set with RFE. PoS and character n-grams are excluded from the feature set.

For LP 2019, the cross-validation score fluctuates with different numbers of features. Besides, accuracy also fluctuates across different context lengths. It has a higher mean difference in accuracy (0.036) than LP-prev (0.012). The possible reason for such fluctuation might be the fewer training data points for LP-2019, resulting in a dramatic difference in the number of data points while varying the context length.

After filtering out highly correlated features, only the performance of the two language pairs, en→lt, and en→ru benefit from such filtering as shown in table 9. This suggests that although some features are highly correlated and might contain redundant information, they are still useful for the classification.

Language	ende	enfi	enlt	enru	deen	fien	guen	kken	lten	ruen	zhen
Feature #	0%	-76.9%	-57.9%	-50%	-75%	-66.6%	-87.5%	-69.2%	-73%	-81.3%	-81.3%
Accuracy	+0	+0	**+0.07**	**+0.02**	+0	-0.01	-0.11	-0.2	-0.03	-0.03	-0.06

Table 9: Difference in classification accuracy after dropping features with VIF greater than 5. Accuracy is measured with a context length of 10 consecutive sentences.

We also conduct the classification with all features, including PoS bi-grams and character ngrams. Initially, we intend to extract the selected ngrams. Thus we adopt two approaches - one-step and three-step feature selection with RFE. For the one-step method, all features are put in the Feature Union, and they are selected

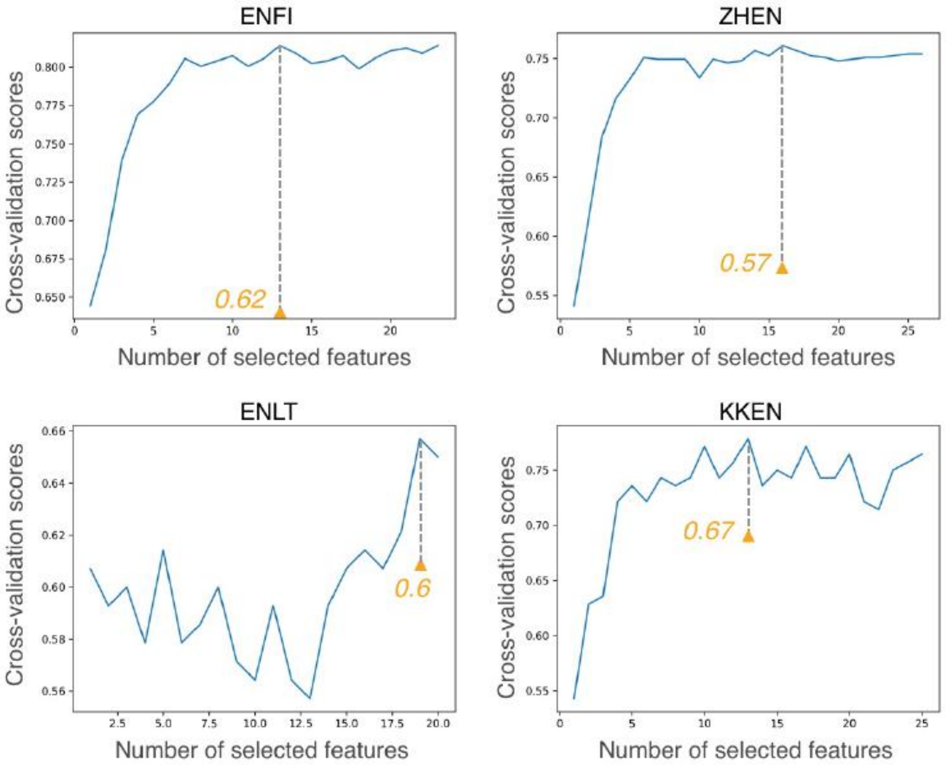


Figure 5: RFECV scores of LP-prev (upper row) and LP-2019 (lower row) in context length of 10. The yellow triangle represents the accuracy on test as indicated in Figure 4. Here we only show selected language pairs from both groups for comparison.

altogether at once. The Three-steps method selects features sequentially. First is to select features from PoS bi-grams and character ngrams. Then all chosen tf-idf ngrams are concatenated together with other features via Feature Union to perform final classification. We only report the results from the one-step method as it yields better accuracy in the MT/HT task. This method achieves the best result of 0.88 for en→lt using only four features, " -"," – ","- ", and "– " which are only spaces with the hyphens. We suspect this is an artifact of the data set that different hyphens are used in MT and HT.

PoS and character n-grams have much helped the classification, especially for LP-2019. For LP-prev, the accuracy for fi→en also increases vastly from 0.56 to 0.72.

However, the representation of tf-idf weights is not comparable with other features, so we do not involve PoS and character n-grams in our further feature analysis in section 5.2.

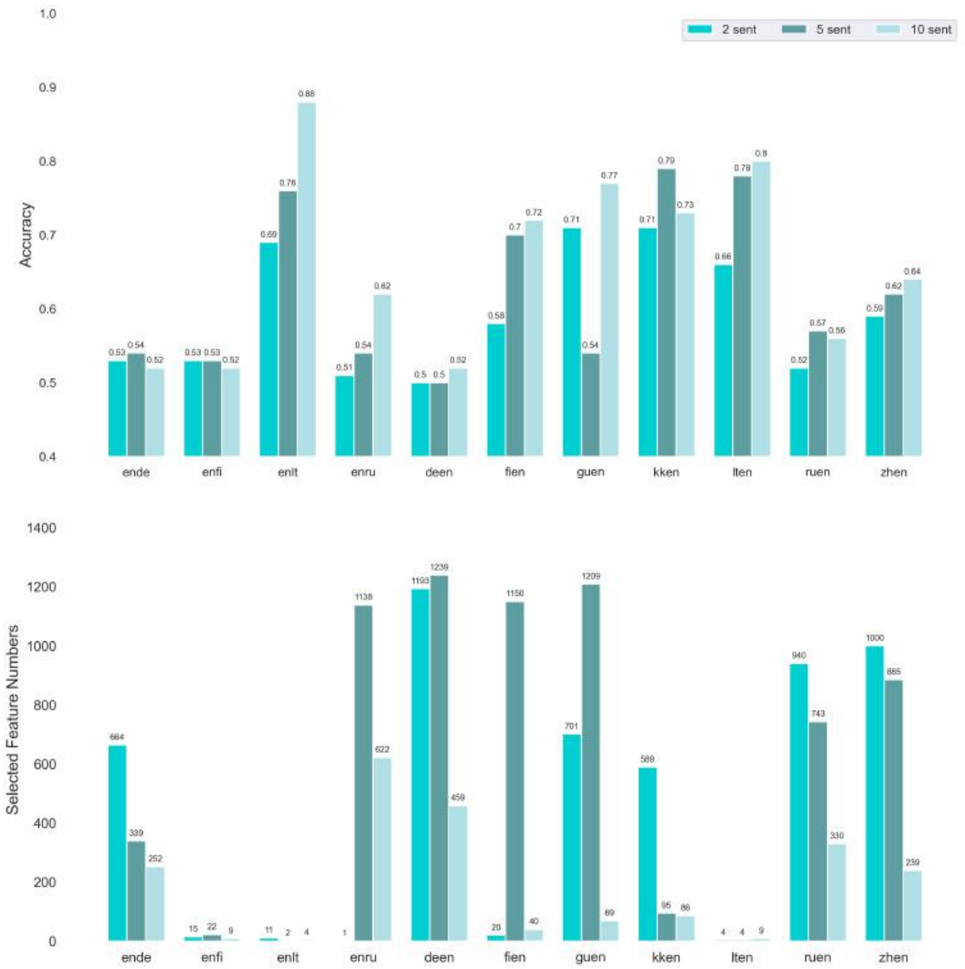


Figure 6: MT-HT classification results using all features including PoS and character ngrams.

5.1.2 Machine translation by DeepL vs. human translation

One reason why LP-2019 has a better performance might be overfitting. Another possibility might be the effect of quality mismatch since the translation quality from previous years might not be on a par with the quality of *newstest2019*. To investigate this, we experiment with the data generated from DeepL.

We hypothesize that classifiers do not suffer from the inferior translation in the training set and could perform better in the unseen data, as mentioned in section 3.1.1. However, as shown in figure 7, only en→ru has better performance while two other language pairs only show slightly better or even worse performance. Classification may depend on the quality of MT systems rather than the quality mismatch between training and testing. The better the MT system is, the harder it gets to distinguish its work from that of HT.

After filtering out highly correlated features, the accuracy drops on average 0.025 across all language pairs and context length. It shows the same trend as in the WMT setting that highly correlated features still provide useful information. Using additional PoS bigrams and character ngrams does not help the classification as well - none of the accuracy scores are above 0.6.

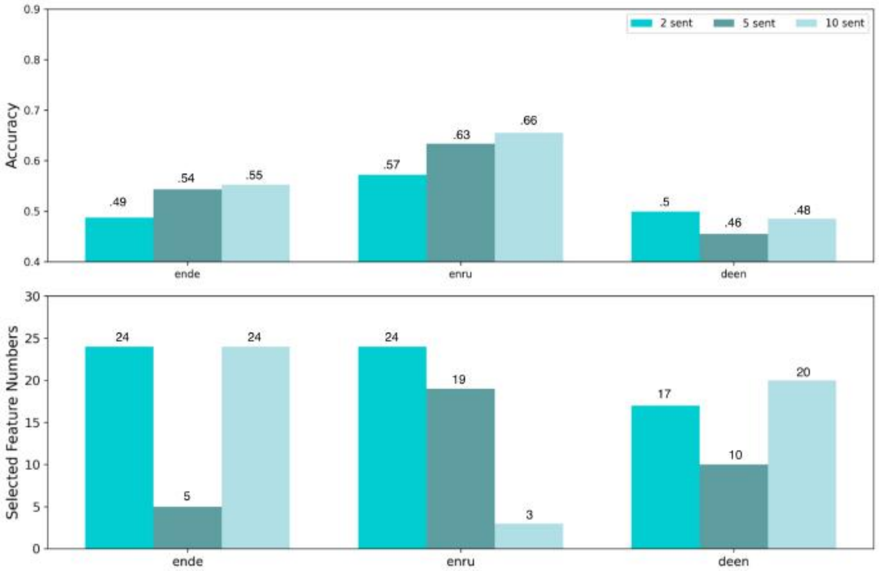


Figure 7: MT-HT classification results using data generated by DeepL. PoS and character n-grams are excluded from the feature set.

5.1.3 Human translation vs. post-edits

There is only one language pair, zh→en, in the HT-PE classification task. We also compare the results of HT-PE with MT-HT in which the MT is provided by the Microsoft NMT system, *Combo-6* (Hassan et al., 2018). The HT-PE task achieves the best accuracy of 0.72 with a context length of 10 using all 26 features. The MT-HT classification task gets a better accuracy of 0.77 with a context length of 10. The best cross-validation scores of both tasks are quite similar, but the discrepancy between cross-validation scores is higher for HT-PE than MT-HT. It also shows that as the context length increases, the accuracy improves.

However, it should be taken into account that MT-based translation is generated differently in both tasks. PE is post-edit based on Google Translate outputs while MT is generated by the best MT system, namely *Combo-6*. We hypothesize that since *Combo-6* is a research-based MT system optimized for quality, its output should be harder to distinguish than MT by Google Translate from HT. Nevertheless, it turns out that the classifier seems to learn better in the MT-HT task. Another thing to note is that since there are only 60 instances (30 instances per label) in the test set with the context length of 10, it might not be representative enough to conclude the classifier's performance.

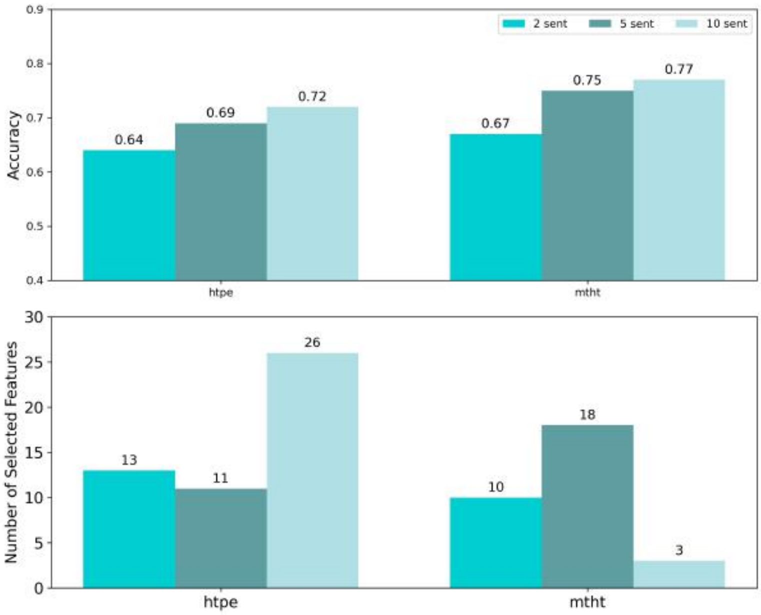


Figure 8: HT-PE and MT-HT classification results on test set with RFE. PoS and character n-grams are excluded from the feature set.

Removing features with higher VIF values greatly decreases the performance for both tasks. The smallest decrease (-0.05) in accuracy happens in MT-HT with a context length of 2. The biggest fall (-0.28) in accuracy is with HT-PE with the context length of 5. Contrarily, including features from PoS bigrams and character ngrams helps to improve the performance as shown in figure 9. HT-PE could reach an accuracy of 0.9 with a context length of 10 consecutive sentences, and MT-HT achieves an accuracy of 0.84 with a context length of 5 consecutive sentences.

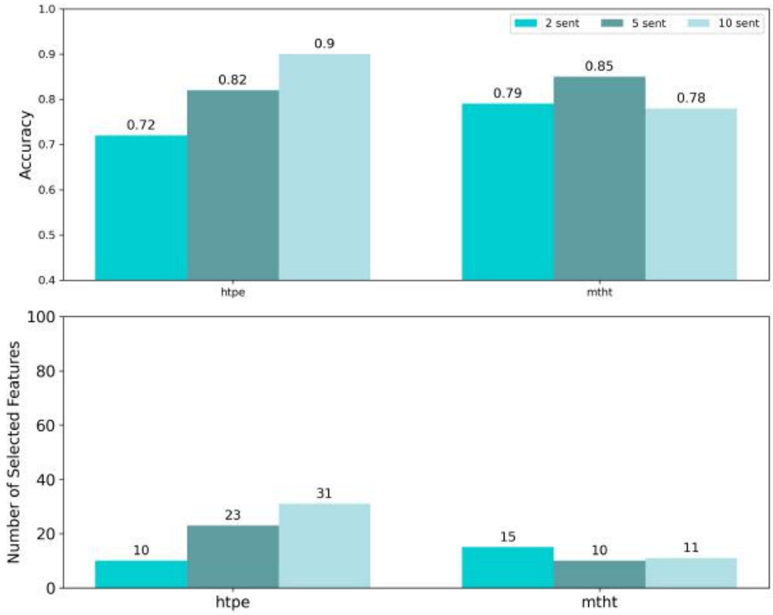


Figure 9: HT-PE and MT-HT classification results using all features including PoS and character ngrams.

5.1.4 Machine translation vs. post-edits

The classification results of MT-PE are presented in figure 10. The highest accuracy of 0.61 is achieved for en→fr with a context length of 10. The selected 23 features are the same as in the context length of 5, while the accuracy increases from 0.56 to 0.62. This indicates that for this translation direction, longer context length indeed helps the classification.

It seems the classifier could somehow distinguish between MT and PE. However, if we look at the cross-validation score of the best performing combination shown in figure 11, the best cross-validation scores are around 0.574 and 0.566 for en→fr and en→nl, respectively. It is counter-intuitive that the test set's accuracy is higher than the cross-validation score, which implies that the testing part might be a better representation of the training data than validation. We suspect this might because data is not randomized well with the sequential split. That is, the first 70% of the data is for training, and the remaining part is for testing.

If we remove features whose VIFs are above 5, the results drop for three language pairs, and none could perform above 0.6. This corroborates the previous findings that although the effect of multicollinearity exists, the classifiers can still learn from those highly correlated features. Table 10 gives an overview of how

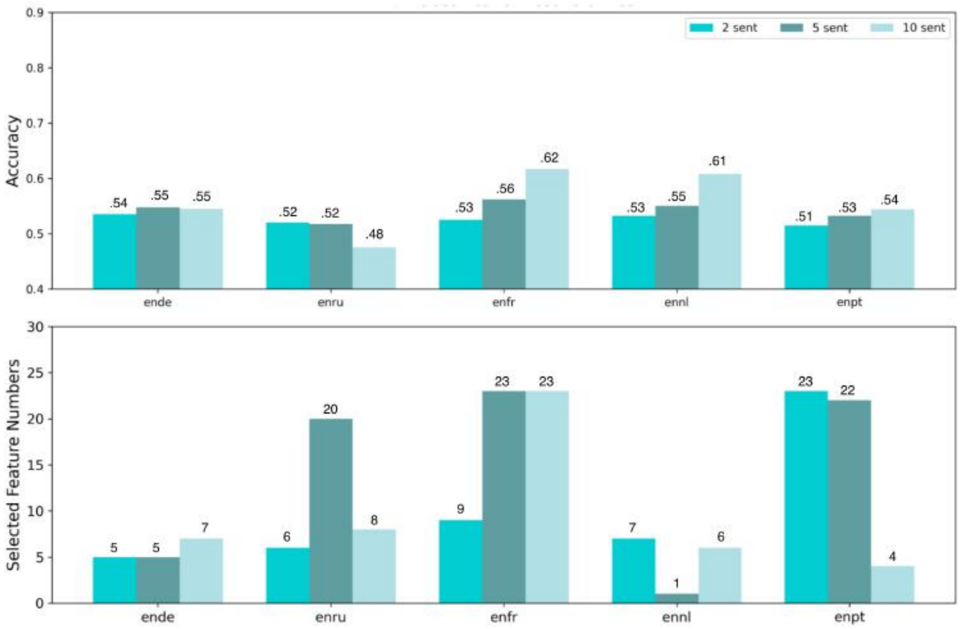


Figure 10: MT-PE classification results on test set with RFE. PoS and character n-grams are excluded from the feature set.

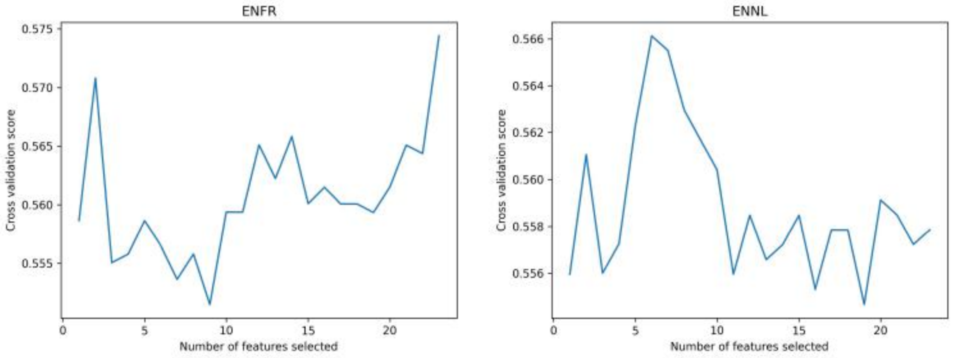


Figure 11: Cross-validation score with different selected feature numbers in the setting of context length of 10.

much accuracy decreases and displays the feature drop rate with context length set at 10. For en→ru and en→pt, removing features does not help improve accuracy but also does not hurt the performance. It suggests that those features are indeed unnecessary for the classification.

Figure 12 shows the classification results using all features. Including PoS and character ngrams greatly helps the classification for en→pt, achieving the best accuracy of 0.8 with context length of 10. It is interesting that for en→nl with context length of 10, it only selects 2 Pos bigrams features, "punct aux" and "verb aux". However, it manages to achieve the best accuracy (0.63) than previous experiments.

Language	ende	enru	enfr	ennl	enpt
Feature #	-71.4%	-37.5%	-73.7%	-66.6%	-50%
Accuracy	-0.02	+0	-0.03	-0.02	+0

Table 10: Difference in classification accuracy after dropping features with high VIFs. Accuracy is measured with a context length of 10.

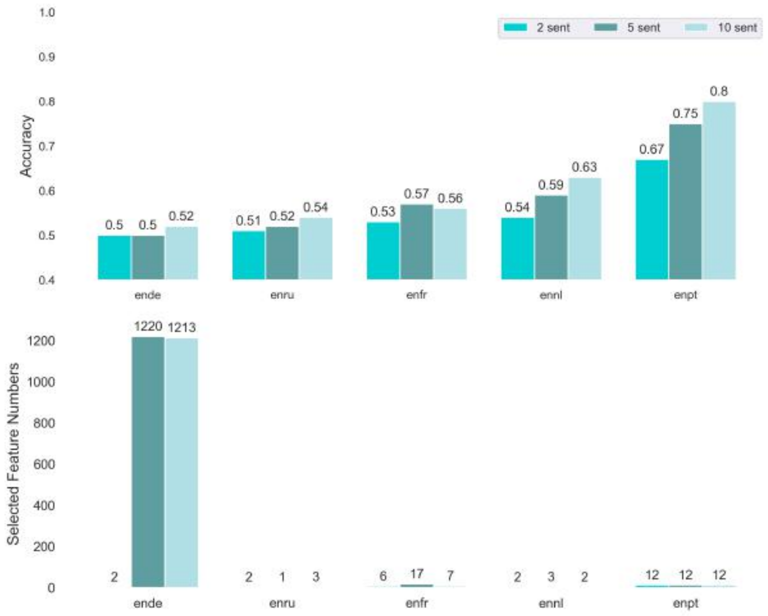


Figure 12: MT-PE Classification results using all features including PoS and character ngrams.

5.2 FEATURE IMPORTANCE

To interpret each translationese hypothesis's importance, two metrics - positional ranking and average coefficients - are computed per feature. Each feature is ranked firstly according to its absolute coefficients. Then the average rank of each feature is calculated across all language and context length combinations. This positional ranking reveals how important each feature is for the classification. That is, when a feature is ranked higher, it helps the classifier more to distinguish between the two labels. On the other hand, average coefficients are calculated per feature as the mean of coefficients across all combinations. The magnitude of this metric suggests how useful the feature is for distinguishing between two translation types.

We compute those two metrics based on the selected features before and after the VIF filter. Nevertheless, the effect of multicollinearity will influence the coefficients' stability and reliability, as mentioned in section 4.2.5, the interpretation derived before applying VIF filter might be problematic. We thus mainly focus on analyzing features derived after the VIF filter.

We visualize the feature's positional ranking and average coefficients with scatter plots. In the scatter plots shown below, each dot represents a feature, and the color indicates its corresponding translationese hypothesis. The size of the dot illustrates this feature's percentage of appearance among all combinations.

5.2.1 Machine translation vs. human translation

Both absolute coefficients ranking and average coefficients imply that **interference** and **normalization** hypotheses are more useful indicators for the classification of MT-HT. In Figure 13, most simplification and explicitation features are removed due to their high VIFs. Although the remaining simplification features still have higher ranks, they are not common among all combinations. Features associated with interference and normalization hypothesis, length ratio and repetition ratio, are among the top rank list with a high presence of around 70% across all combinations. Looking at the average coefficients, as shown in Figure 14 (right), both features have greater magnitude in coefficients than most of the other features. Another feature, passive verb ratio, has a relatively higher percentage of presence (48%) and considerable magnitude in coefficients.

Table 11 shows the weighted coefficients per translationese hypothesis. The weighted coefficient is derived from multiplying the feature coefficients by the percentage of presence. Judging from the weighted coefficients, we could confirm that **interference** and **normalization** have greater magnitude, thus they are distinctive indicators in distinguishing between MT and HT. We also calculate another metric by taking the mean of the absolute coefficients for each translationese hypothesis. It shows that normalization is the most distinctive translationese characteristics in MT-HT, followed by interference.

We also further inspect the values of interference and normalization features to validate whether our hypothesis stated in section 4.1 is true. Figure 15 shows that

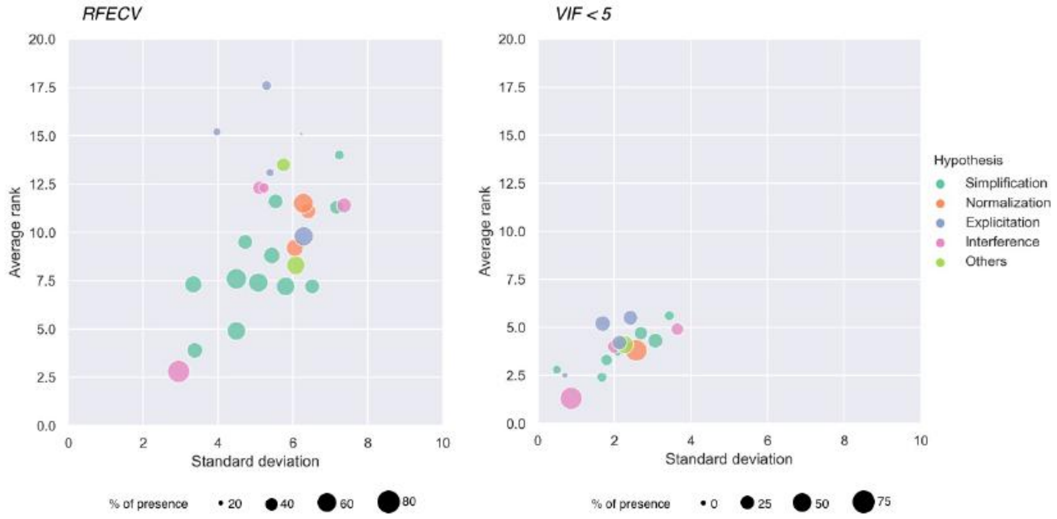


Figure 13: Average rank and standard deviation of features used in MT-HT classification. Left: feature selected after RFE. Right: features whose VIFs are < 5.

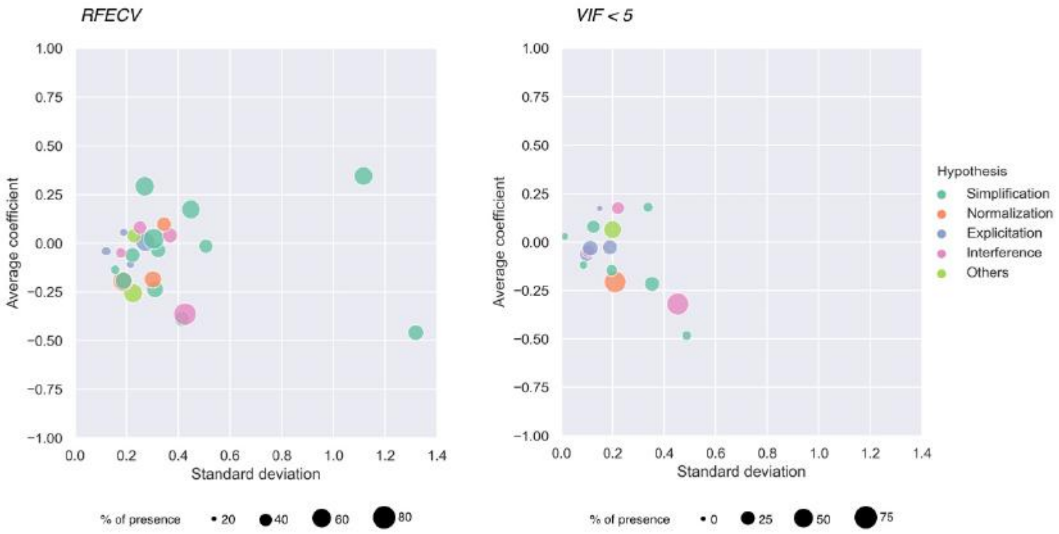


Figure 14: Average coefficients and standard deviation of features used in MT-HT classification. Left: feature selected with RFECV. Right: features whose VIFs are < 5.

Hypothesis	Weighted coef	Abs. coef
Simplification	-0.015	0.178
Normalization	**-0.075**	**0.209**
Explicitation	-0.008	0.073
Interference	**-0.068**	**0.18**
Others	0.031	0.064

Table 11: The weighted coefficients and average absolute coefficients for MT-HT.

the length ratio is lower for MT in most language pairs, which confirms our hypothesis that MT has more interference from the source languages than HT. Nonetheless, the values of the repetition ratio contradict our hypothesis. They are higher in HT, implying more repeated content word usage. Notwithstanding, we suspect this might be related to the greater amount of content words used in HT.

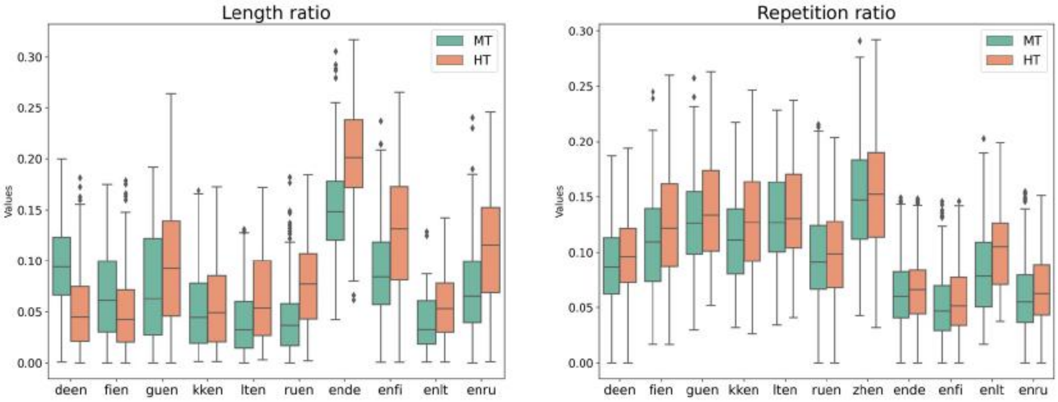


Figure 15: Box-plots of interference (left: length ratio) and normalization (right: repetition) features derived from the training set of MT-HT.[1]

5.2.2 Human translation vs. post-edits

Since there is only one language pair involved in HT-PE, there are only three combinations (i.e., context length of 2, 5, and 10) to calculate the metrics of feature importance. Looking at Figure 16 and Figure 17, most explicitation and normalization features are eliminated due to the high VIFs, or they are not presented in the scatter plot with very low presence. Only three features - MDD, PoS perplexity, and perplexity - remain on the plot. The **simplification** feature, MDD, has the highest rank of 1, followed by the **interference** feature of PoS perplexity.

In table 12, we could see that **interference** has the highest average of absolute coefficients of 0.3, followed by **Others** with an average of absolute coefficients of 0.239. Nevertheless, this result should be regarded with caution because there is only one feature in this category. Nonetheless, weighted coefficients show that interference has the greatest magnitude in coefficient, followed by simplification. It is interesting that for this task, compared with MT-HT and MT-PE, totally different

---

[1] We exclude the presentation of zh→en in length ratio because the difference of character counts between English and Chinese are not in the approximate range as the other language pairs due to the nature of Chinese character segmentation. This situation inflates the y-axis in the plot.

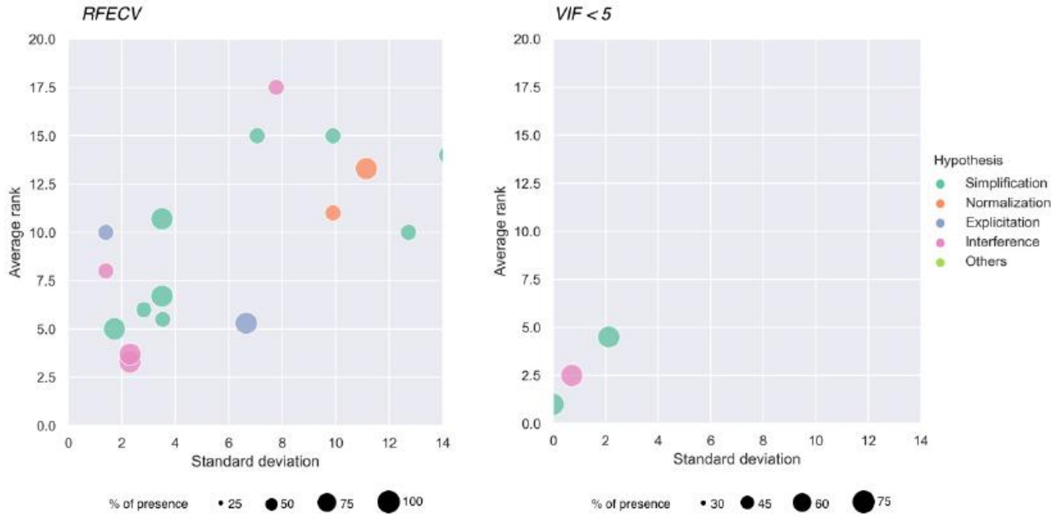


Figure 16: Average rank and standard deviation of features used in HT-PE classification. Left: feature selected with RFECV. Right: features whose VIF are < 5.

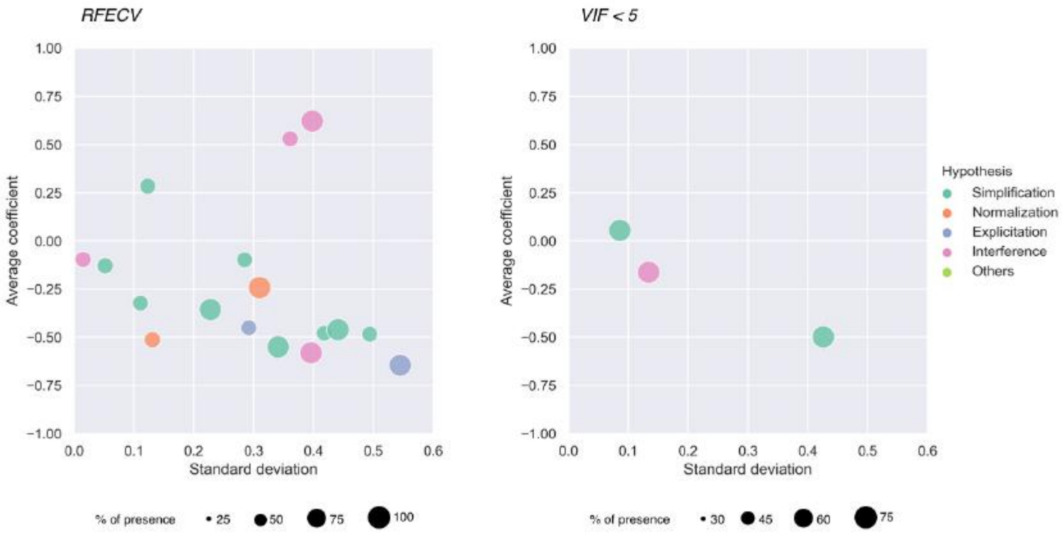


Figure 17: Average coefficients and standard deviation of features used in HT-PE classification. Left: feature selected with RFECV. Right: features whose VIF are < 5.

subsets of features are selected after the VIF filter. We assume this might be because of the relatively small data size, which leads to an unexpected bias. Also, filtering out features with high VIFs dramatically hurts the performance in this task; thus whether the contributions of the remaining features are positive for this task are doubtful. Although the data points might be too few to draw a general conclusion about the feature importance, we could still see a trend that features belonging to the interference hypothesis seem to be strong indicators across different classification tasks.

Hypothesis	Weighted coef	Abs. coef
Simplification	**-0.11**	0.22
Normalization	-	-
Explicitation	-0.029	0.098
Interference	**-0.127**	**0.3**
Others	-0.079	**0.239**

Table 12: Average coefficients and percentage of presence per translationese hypothesis for HT-PE.

Figure 18 shows the values of simplification and interference features in MDD and the perplexity difference of PoS sequence. MDD, as expected, HT has a more complex syntactic structure (i.e., a higher degree of MDD) than MT and HT. Perplexity difference is also higher in HT than MT and PE, signifying it is less prone to interference from source languages, which matches our hypothesis. Nevertheless, PE is subjected to a more considerable degree of interference than MT. We presume the reason might be that PE is primed by Google Translate, whose quality is not comparable with the quality-oriented Microsoft NMT system.

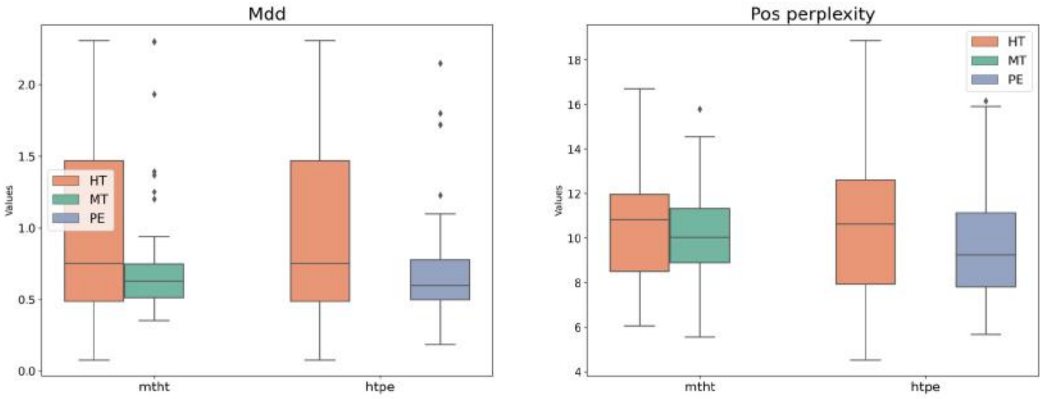


Figure 18: Box-plots of simplification (left: MDD) and interference (right: PoS perplexity) features derived from the training set with the context length of 10.

5.2.3 Machine translation vs. post-edits

Looking at Figure 19 (right), the more apparent features are from the categories of interference and explicitation. The length ratio belonging to the interference hypothesis again is the most common and evident feature. As evident from Figure

20, it has greater magnitude in coefficients than the rest of the features, signifying its importance for the classification.

Three explicitation features are also commonly presented in almost half of the combinations but only one of them has bigger magnitude (0.205) in coefficients. Nonetheless, one feature from the normalization hypothesis - repetition - is particularly noticeable. It has a high rank of 3 and great magnitude (-0.151) in coefficients.

To sum up, our experiments show that **interference** and **explicitation** are prominent attributes to distinguish between MT and PE.

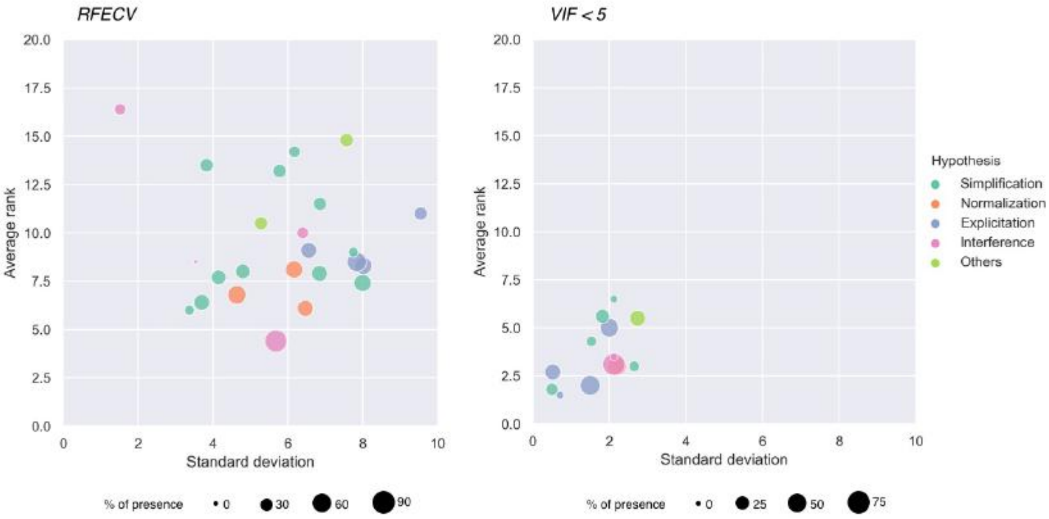


Figure 19: Average rank and standard deviation of features used in MT-PE classification. Left: feature selected with RFECV. Right: features whose VIF are < 5.

Table 13 shows the weighted coefficients per translationese hypothesis. Among all, **interference** and **explicitation** have bigger magnitude in weighted coefficients. When looking at the mean of absolute coefficients, it shows normalization is the most dominant characteristic, followed by explicitation. This result is, however, not consistent with our previous interpretation. We suspect this might be because of the way of how the average of absolute coefficients is calculated that it does not take the percentage of presence of each feature into account. It might create a bias towards features with high coefficients but low presence.

Figure 21 shows the values of the most prominent features - length and repetition ratio. Length ratio is lower in MT than in PE, indicating that MT is influenced to a more considerable degree by the source languages, as indeed we hypothesized. However, the average function word does not show a consistent direction towards any label.

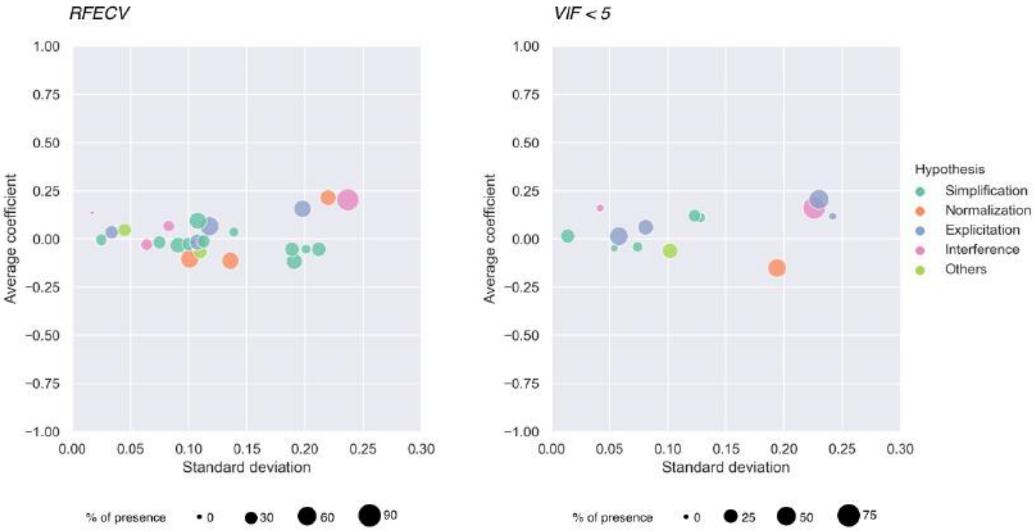


Figure 20: Average coefficients and standard deviation of features used in MT-PE. Left: feature selected with RFECV. Right: features whose VIF are < 5.

Hypothesis	Weighted coef	Abs. coef
Simplification	0.005	0.093
Normalization	-0.028	**0.242**
Explicitation	**0.043**	**0.1**
Interference	**0.035**	0.092
Others	-0.025	0.062

Table 13: The weighted coefficients and average of absolute coefficients for MT-PE.

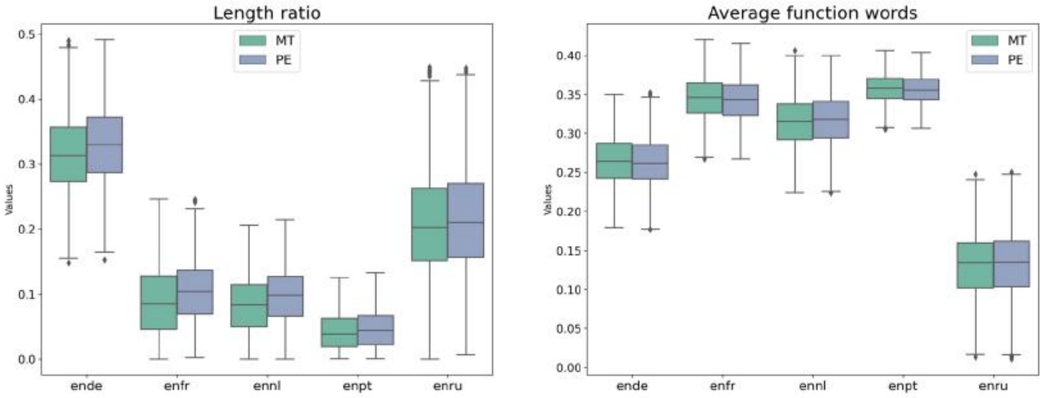


Figure 21: Box-plots of interference (left: length ratio) and explicitation (right: average function words) features derived from the training set with the context length of 10.

5.3 DIRECT ASSESSMENT AND EDIT DISTANCE

As a previous study points out, there is a negative relationship between the classification accuracy and the quality of the MT systems (Aharoni et al., 2014). Figure 22 (left) shows the Pearson correlation and 95% confidence region of the DA score of the best MT system against the classification accuracy with a context length of 10. This negative relationship is significant (*r(9)=-.82, p=0.0018*), corroborating the previous findings. This negative relationship is also significant when the classification is based on the context length of 2 (*r(9)=-.81, p=0.0025*) and context length of 5 (*r(9)=-.68, p=0.022*).

We also hypothesize that if the amount of edits required to change MT output into PE is fewer (i.e., PEs are more similar to MT), it is harder to distinguish between the two, resulting in a positive relationship between the two metrics. Figure 22 (right) shows the Pearson correlation and 95% confidence region of the Translation Edit Rate (TER) against the classification accuracy with a context length of 10. This relationship is strong but not significant, which might also be the case with the context length of 2. However, this positive relationship is strong and significant (*r(3)=.95, p=0.014*) with the context length of 5.

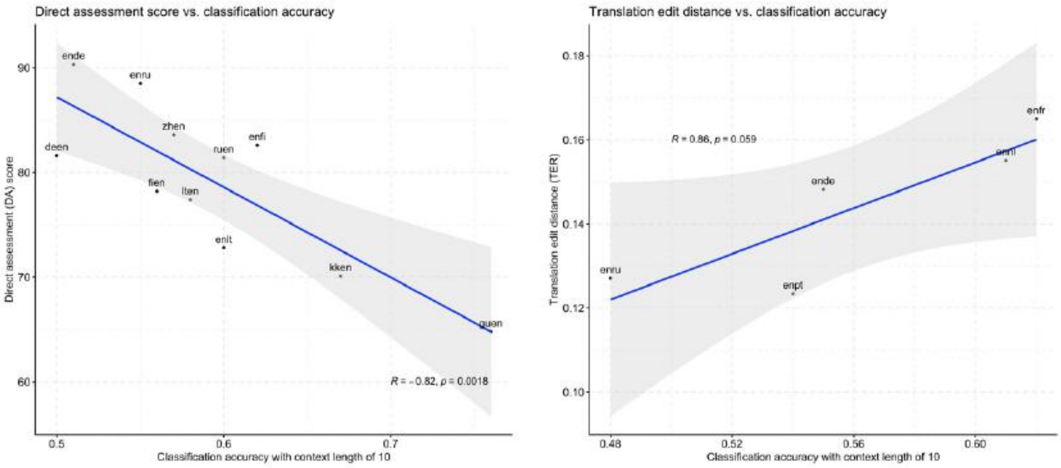


Figure 22: Pearson correlation between and classification accuracy with context length of 10 and the direct assessment score (left) and translation edit rate (right).

6 CONCLUSION

Studies have shown that readers still prefer human translation (HT), even over translations produced by state-of-the-art Machine Translation systems. What often neglected when measuring translation quality in MT as compared to HT are lexical and syntactic differences. With this research, we aim to fill in this research gap and focus on identifying the characteristics that enable us to distinguish between three types of commonly available translations - MT, PE, and HT.

We have employed a machine-learning algorithm and conducted three binary classification tasks (i.e., MT-HT, MT-PE, and HT-PE) using the linguistic attributes inspired by translation studies. Those linguistic features model the phenomenon of *translationese* in four aspects - simplification, explicitation, normalization, and interference. It achieves the mean accuracy of 0.55 for MT-HT, 0.56 for MT-PE, and 0.72 for HT-PE with a context length of 10 consecutive sentences. We analyzed the importance of each translationese aspect through the absolute coefficients ranking and average coefficients. The effect of multicollinearity is also taken into account, and highly correlated features are eliminated for reliable and stable coefficients. Finally, we conducted the Pearson correlation test, and it confirms the previous finding (Aharoni et al., 2014) that classification accuracy has a strong and significant inverse relationship with the DA score. Although the classifier's performance is only slightly better than the chance level for two of the three tasks, this study still sheds light on the unique translationese phenomenon exhibited nowadays among the currently flourishing translation types.

Based on the absolute coefficient ranking and average coefficients, we conclude that for HT-PE, machined-based translation (i.e., PE) can be distinguished from HT with relatively good accuracy, while for MT-HT, the accuracy is dependent on the translation quality of the MT systems as there is a significantly negative relationship between classification accuracy and DA score. (RQ1) However, we should treat the result of HT-PE with caution as the data amounts are few, which has possibly caused overfitting.

Our experiment result also shows that different aspects of translationese are useful for distinguishing machine-based translation from HT. Features associated with **interference** and **normalization** help the classifier to discriminate MT from HT whereas features indicative of **interference** and **simplification** are among the most useful indicators for HT-PE. (RQ2) **Interference** seems to be the common and primary characteristic, however, it is difficult to make a generalization of the importance of any translationese aspects as the datasets in each task are different. Finally, features which exemplify the phenomenon of **interference** and **explicitation** have the best discriminating power between the two machine-based translation (i.e., MT and PE). (RQ3) We also find that features indicative of interference (length ratio and PoS perplexity) reveal that the source languages indeed influence machined-based translations to a larger degree than HT.

Given the previous findings, some suggestions for further research are worth exploring in the future. Firstly, it might be beneficial to implement other features inspired by previous studies. For example, Kendall's tau distance (Birch and Osborne, 2011; Toral and Sánchez-Cartagena, 2017) can be used to measure the similarity between the word alignments of translation and reference texts. Moreover, since most of our features are modeled at the lexical level, incorporating more syntactic features which have proved to have considerable predictive power in translationese detection (Kunilovskaya and Kutuzov, 2017) might be useful. Secondly, since dropping the highly correlated features hurts the classifier, the trade-off between one feature's weights and multicollinearity degree could also be further investigated.

Thirdly, we use different datasets in which translations are not only generated differently but in different domains. It further affects the interpretation of the generality of the translationese aspects across translation types. For example, the MT-PE task uses data from IT and legal domains, which are very different from the news domain in MT-HT. Future works could consider using a more open domain dataset, such as eScape (Negri et al., 2018), which contains several corpora belonging to various domains. Finally, for the methodology, a potential approach could incorporate translationese features into fine-tuned BERT (Devlin et al., 2018). One possible approach is to convert translationese features into embeddings via a multilayer perceptron (MLP) network and combine it with BERT embeddings.

To sum up, our work explores the phenomenon of translationese in the commonly-used translation types by integrating translation studies with a computational approach. We confirm that different translations demonstrate various linguistic attributes, exhibiting different levels and dimensions of so-called translationese. However, under the scope of our work, it is difficult to make a general conclusion on the importance of any translationese aspect due to the limitation of the datasets used.

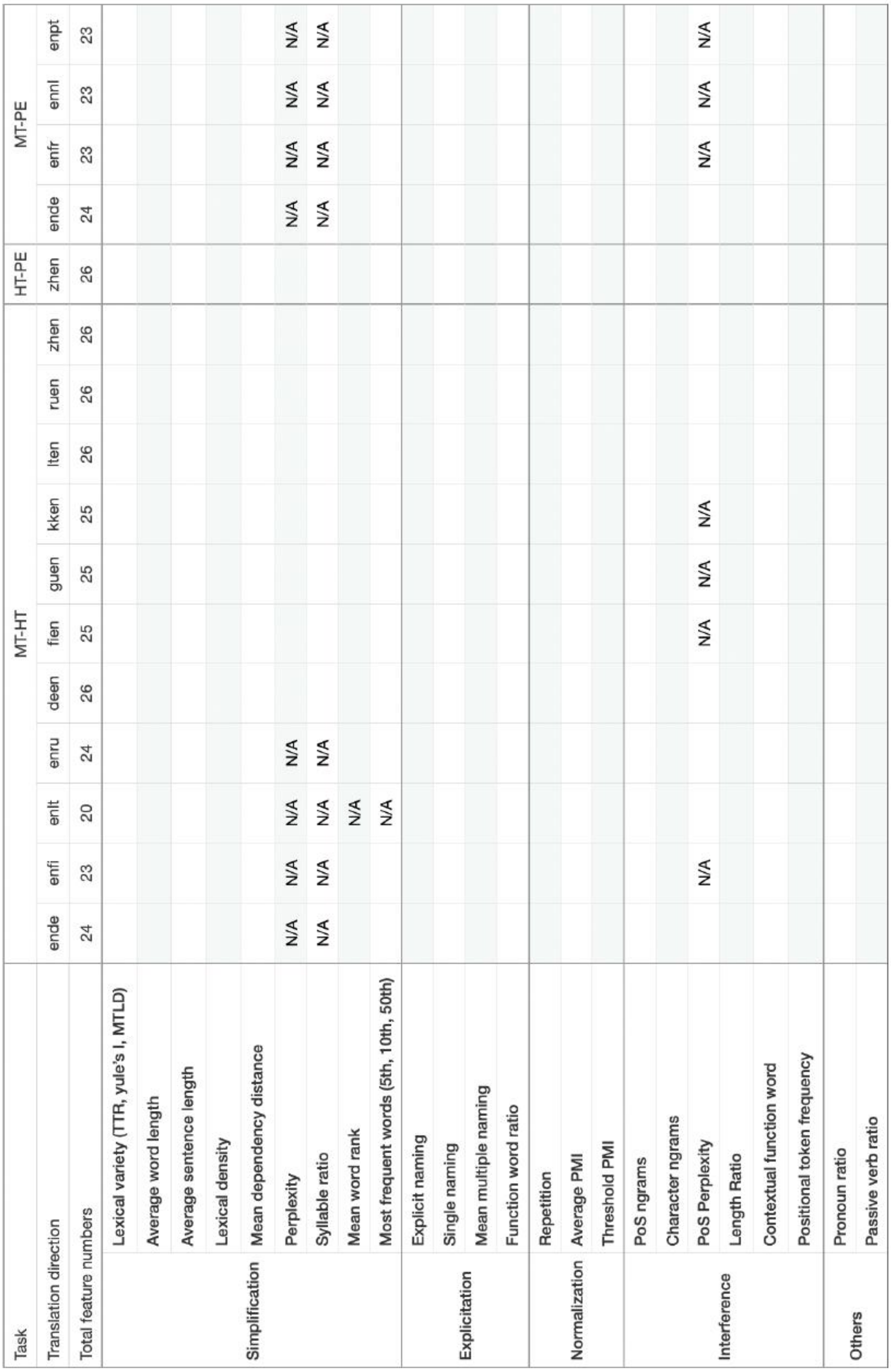
Lastly, we would like to mention that all the code and data used in our experiments are available on Github[1].

---

[1] `https://github.com/yuwenchen31/translation_detection`

Appendices

A IMPLEMENTED FEATURES PER LANGUAGE PAIR



Task		MT-HT											HT-PE	MT-PE			
Translation direction		ende	enfi	enlt	enru	deen	fien	guen	kken	lten	ruen	zhen	zhen	ende	enfr	ennl	enpt
Total feature numbers		24	23	20	24	26	25	25	25	26	26	26	26	24	23	23	23
Simplification	Lexical variety (TTR, yule's I, MTLD)																
	Average word length																
	Average sentence length																
	Lexical density																
	Mean dependency distance																
	Perplexity	N/A	N/A	N/A	N/A									N/A	N/A	N/A	N/A
	Syllable ratio	N/A	N/A	N/A	N/A									N/A	N/A	N/A	N/A
	Mean word rank			N/A													
	Most frequent words (5th, 10th, 50th)			N/A													
Explicitation	Explicit naming																
	Single naming																
	Mean multiple naming																
	Function word ratio																
Normalization	Repetition																
	Average PMI																
	Threshold PMI																
Interference	PoS ngrams																
	Character ngrams																
	PoS Perplexity		N/A				N/A	N/A	N/A						N/A	N/A	N/A
	Length Ratio																
	Contextual function word																
	Positional token frequency																
Others	Pronoun ratio																
	Passive verb ratio																

Table 14: Implemented features in each translation direction. N/A represents this feature is no provided.

B BOX-PLOT OF FEATURE VALUES

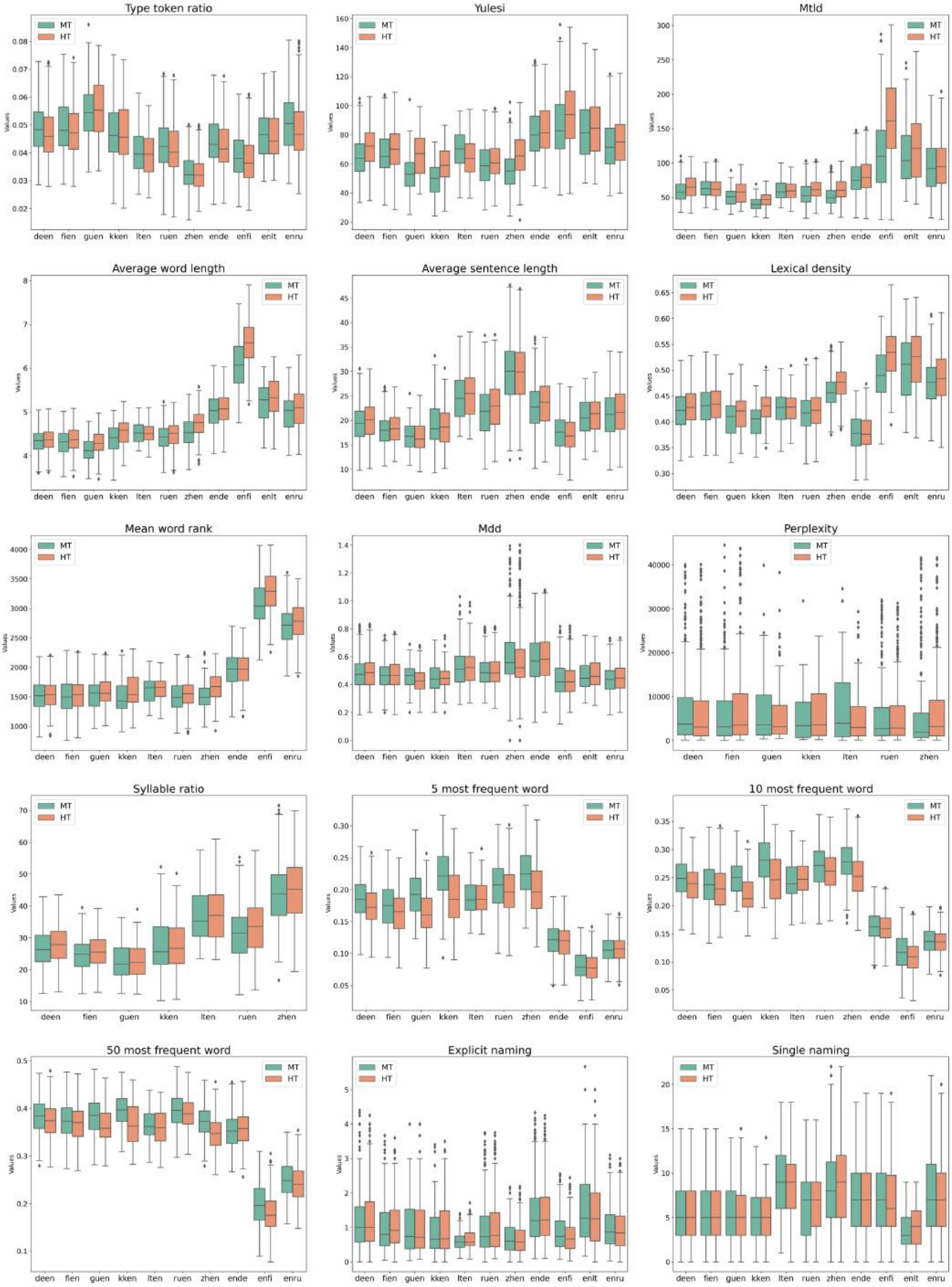


Figure 23: Feature values computed on the training set of MT-HT.

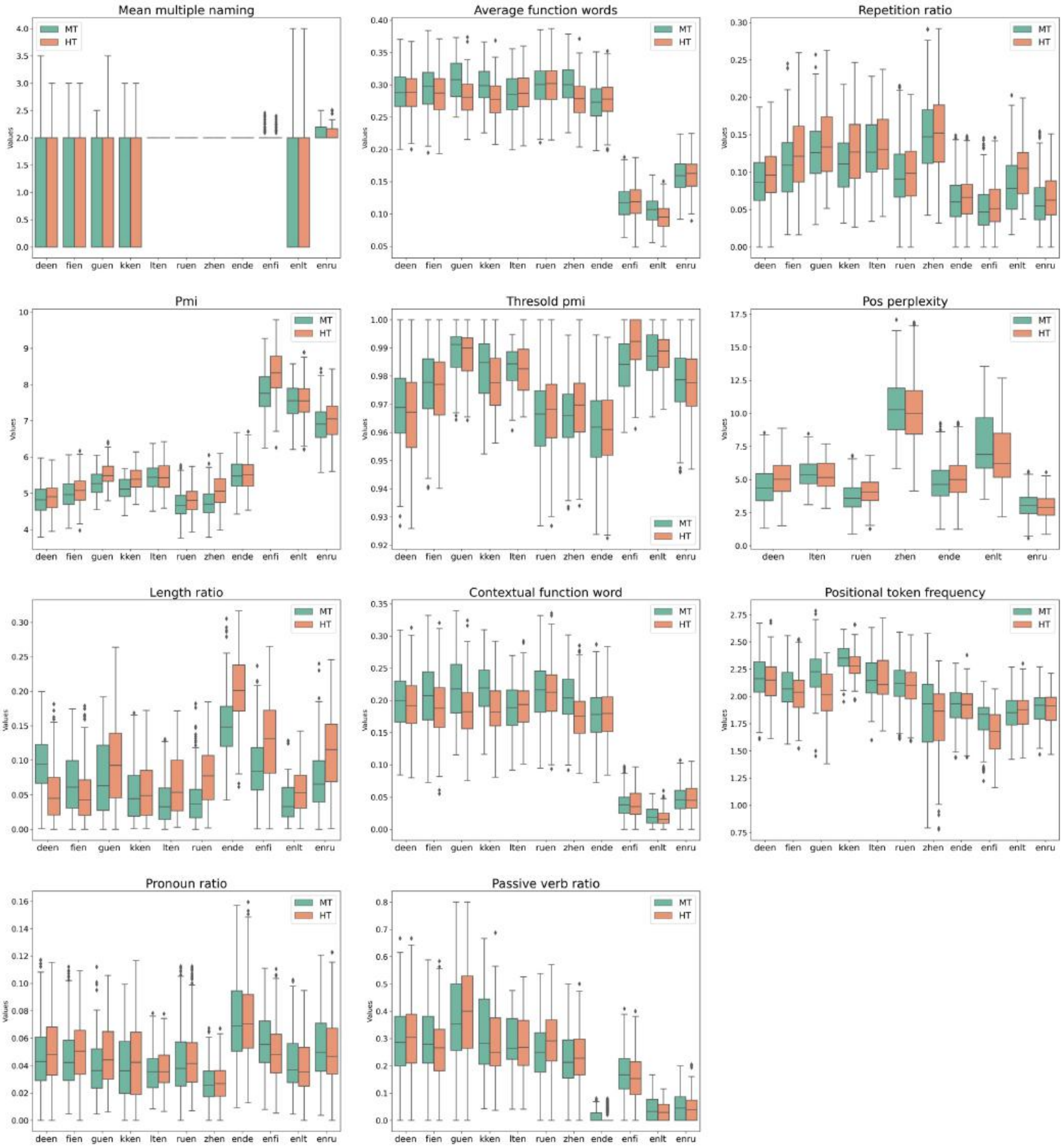
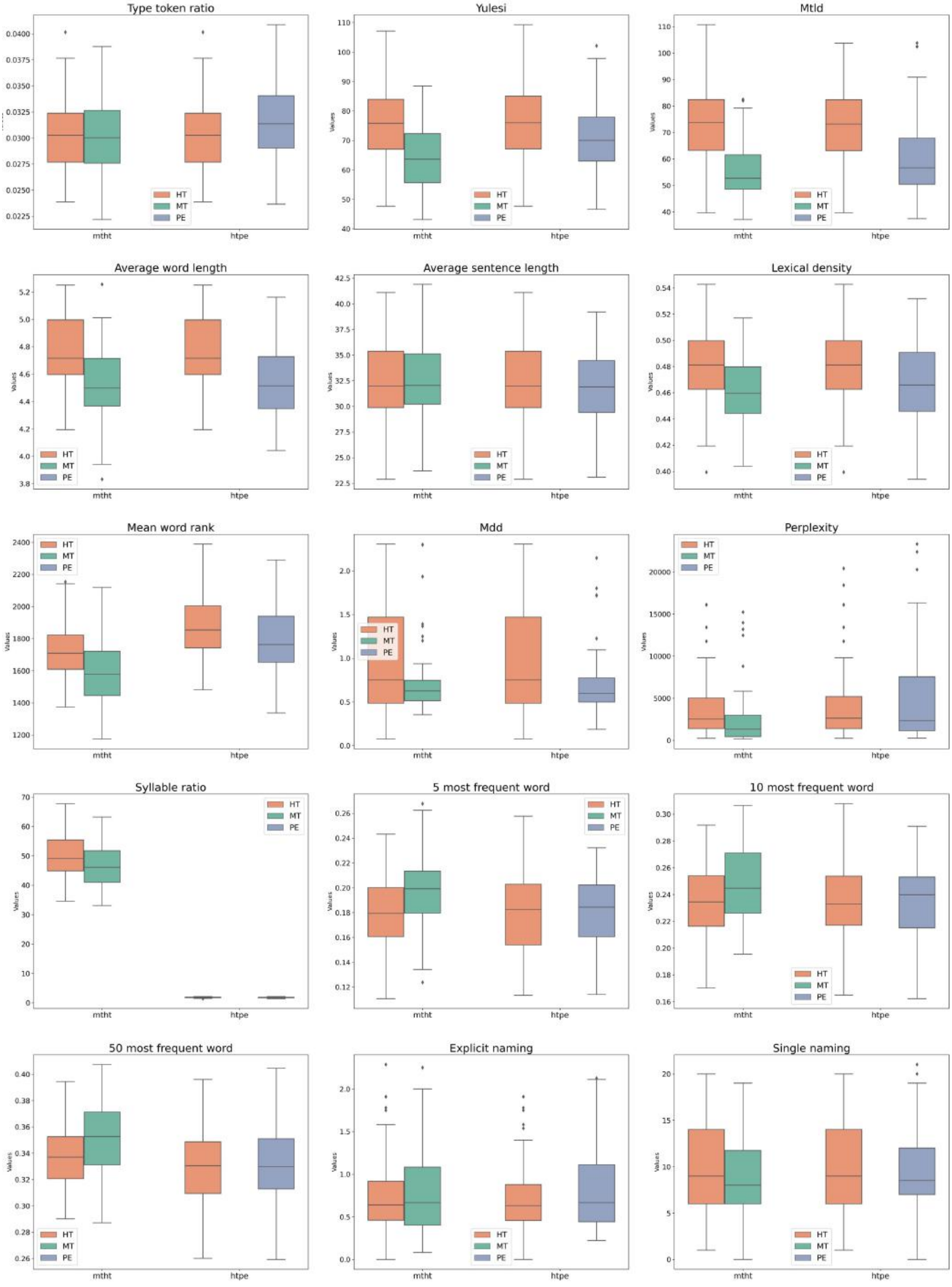


Figure 24: Feature values computed on the training set of MT-HT.



Type token ratio

Yulesi

Mtld

Average word length

Average sentence length

Lexical density

Mean word rank

Mdd

Perplexity

Syllable ratio

5 most frequent word

10 most frequent word

50 most frequent word

Explicit naming

Single naming

Figure 25: Feature values computed on the training set of HT-PE.

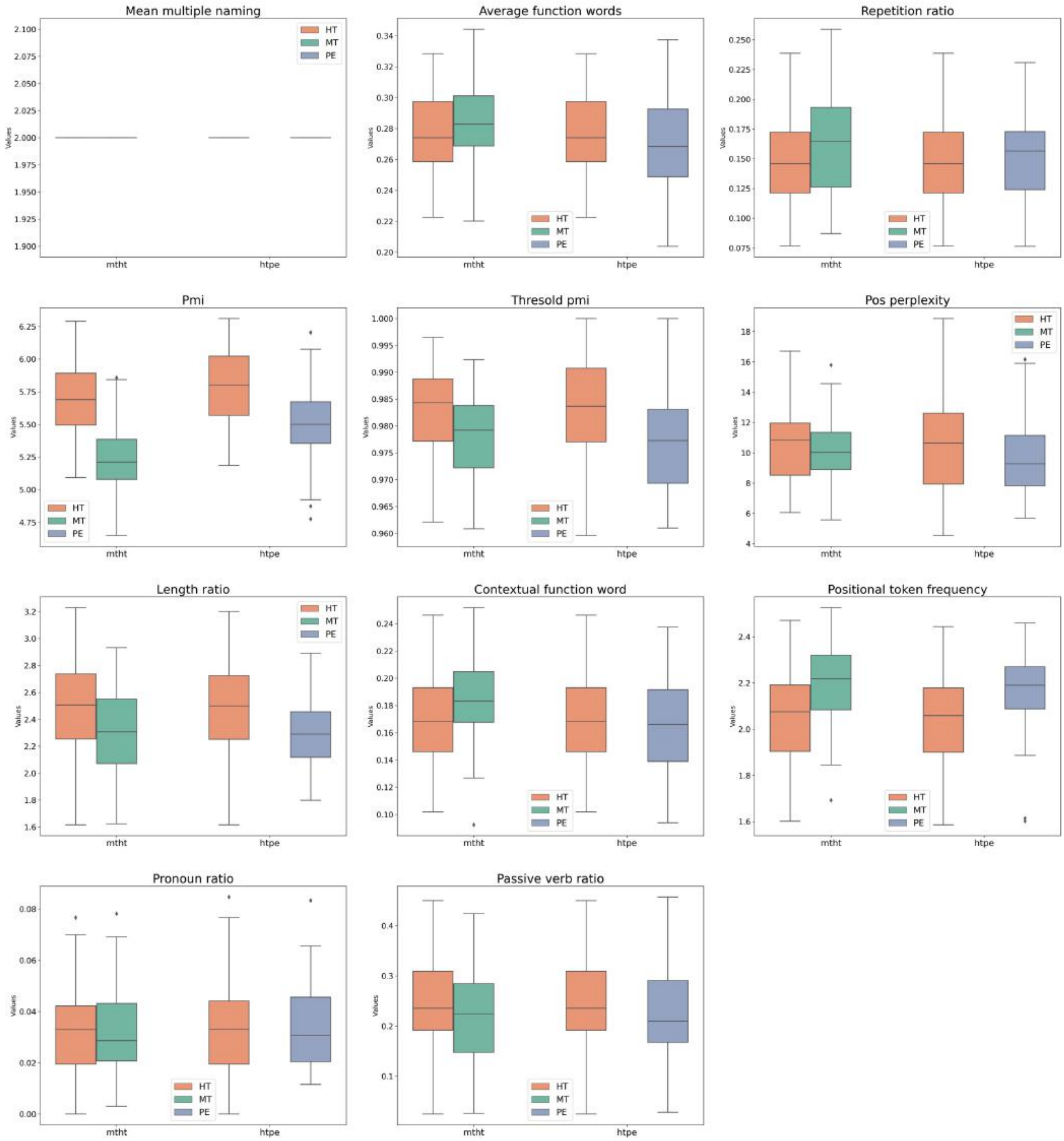


Figure 26: Feature values computed on the training set of HT-PE.

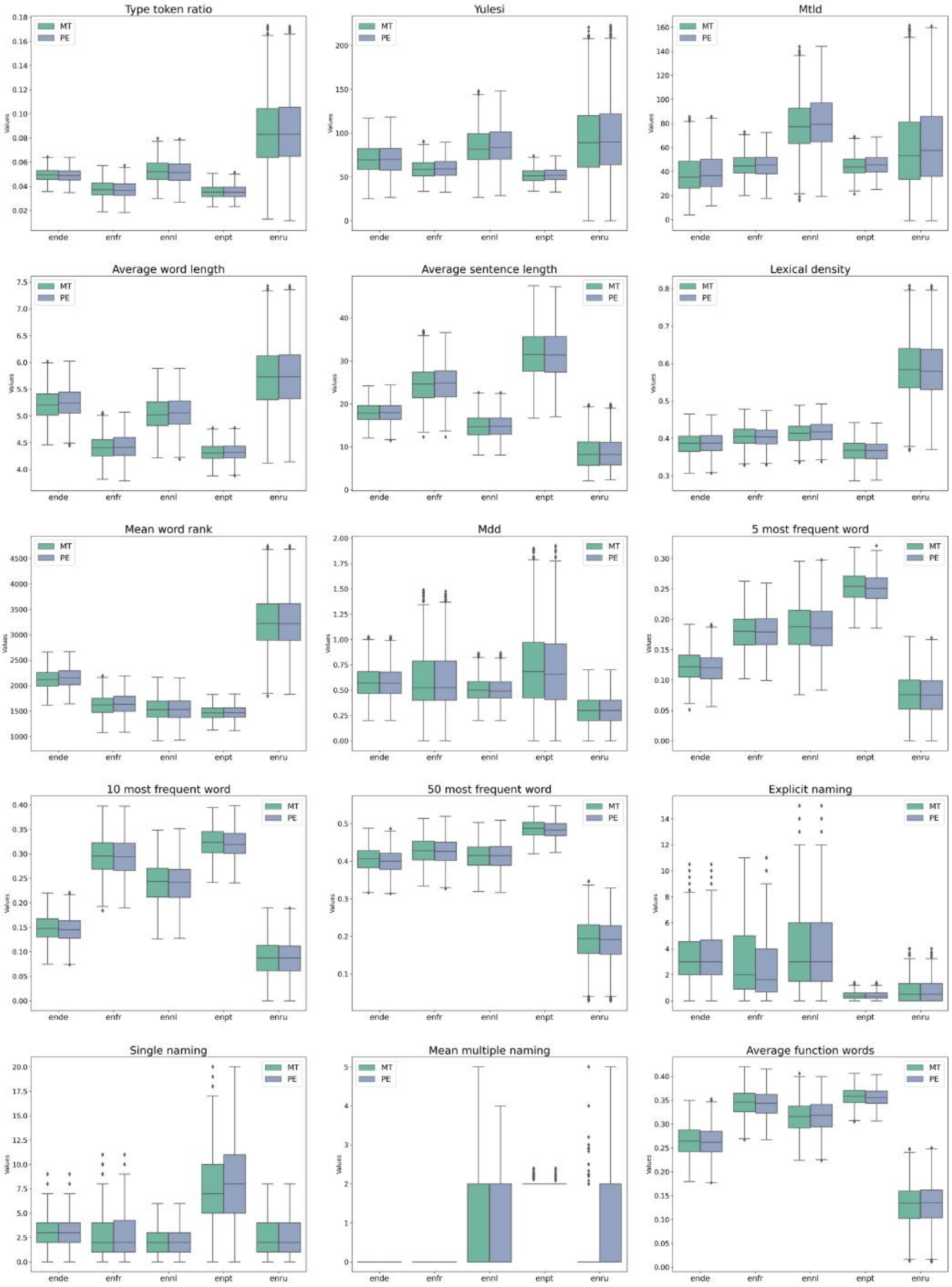


Figure 27: Feature values computed on the training set of MT-PE.

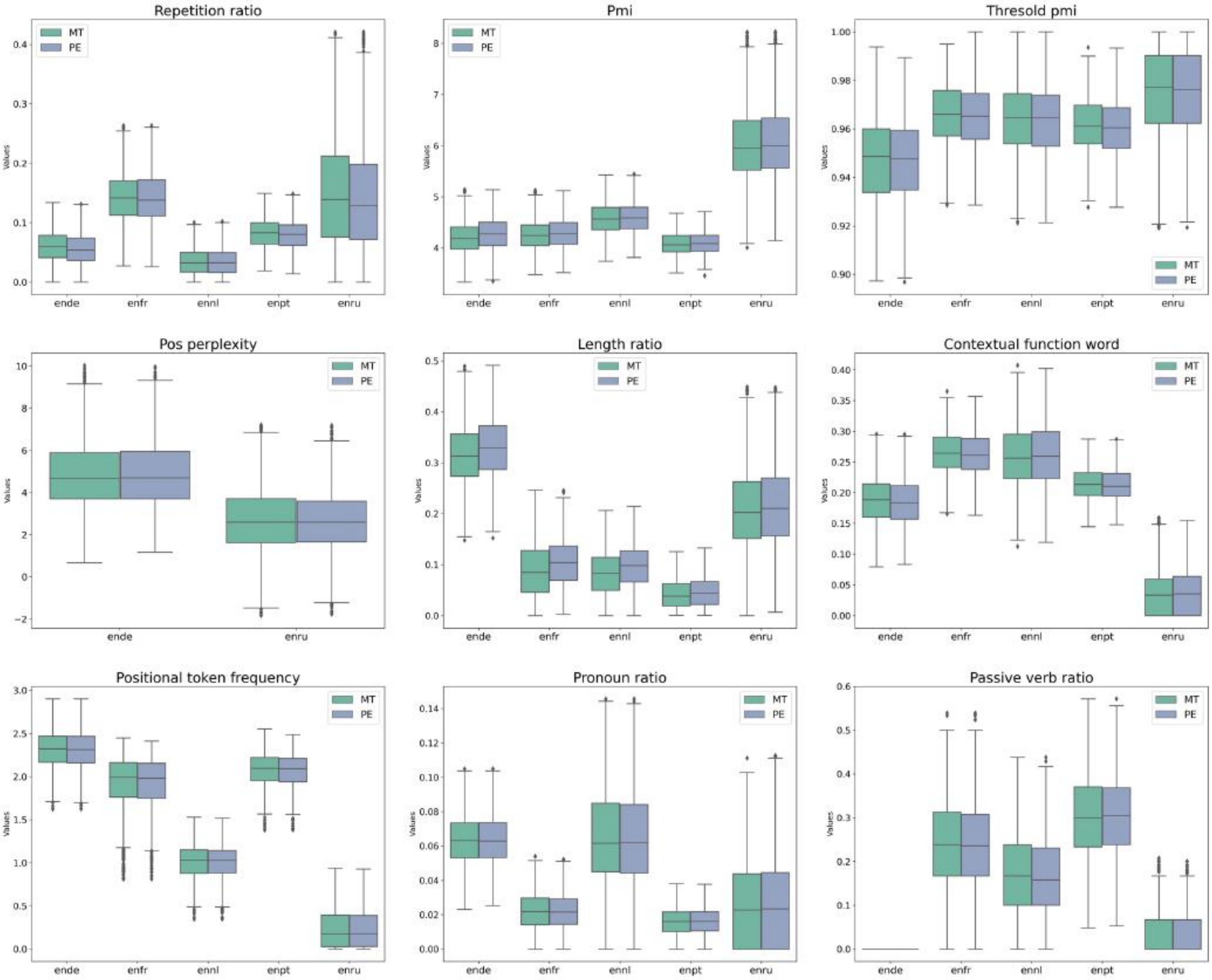


Figure 28: Feature values computed on the training set of MT-PE.

C CLASSIFICATION RESULTS USING FEATURES WHOSE VIF < 5

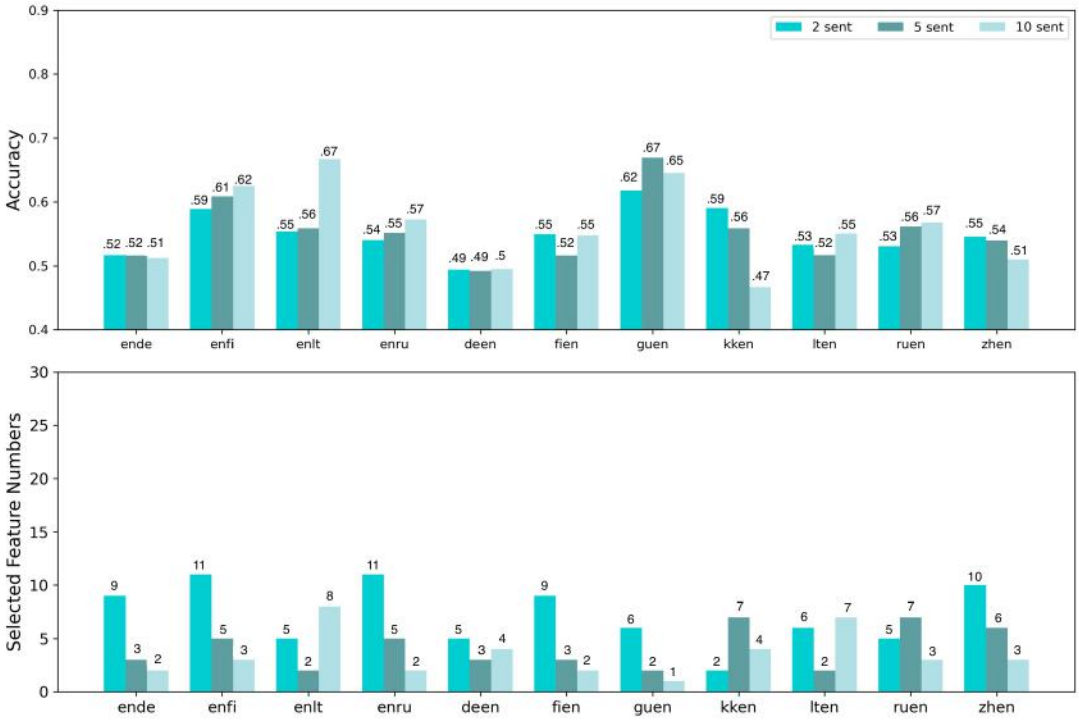


Figure 29: MT-HT Classification results using features whose VIFs are below 5.

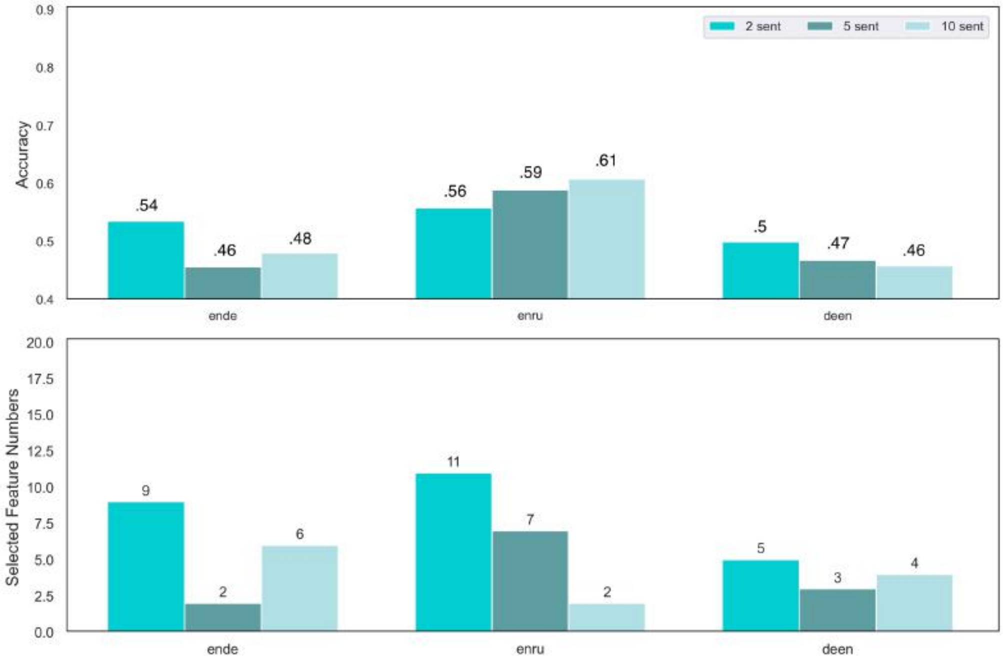


Figure 30: MT-HT classification results using data generated by DeepL with features whose VIF < 5.

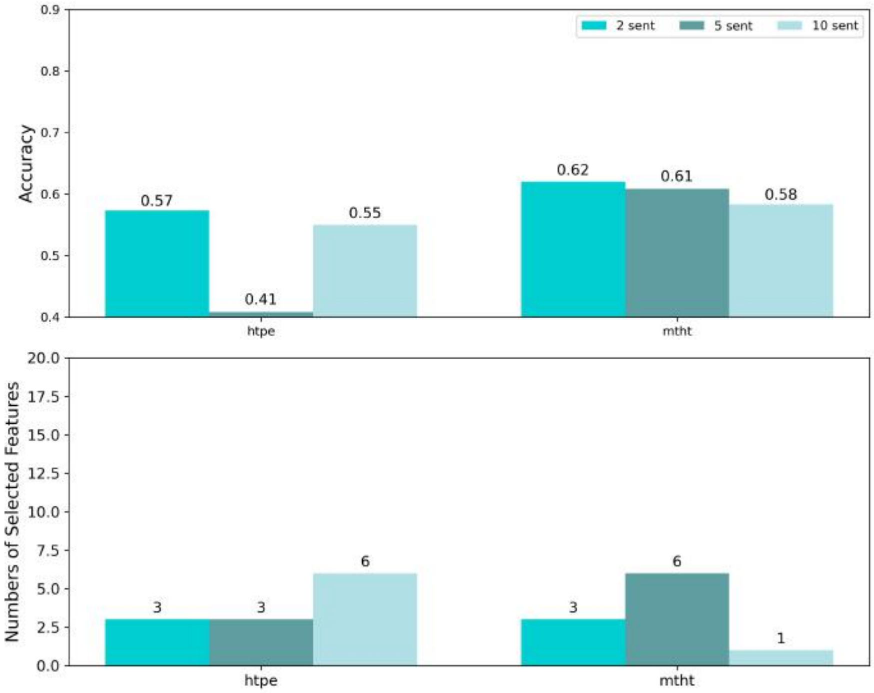


Figure 31: HT-PE and MT-HT Classification results with features whose VIFs are below 5.

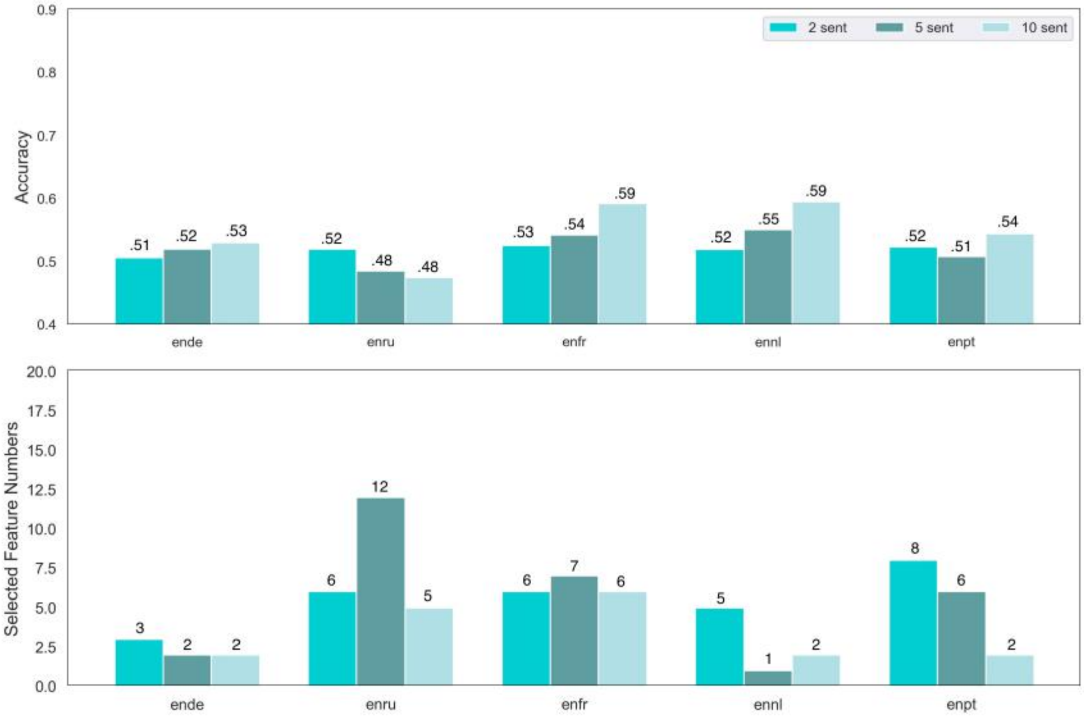


Figure 32: MT-PE Classification results with features whose VIFs are below 5.

D CROSS-VALIDATION SCORES DURING RFE

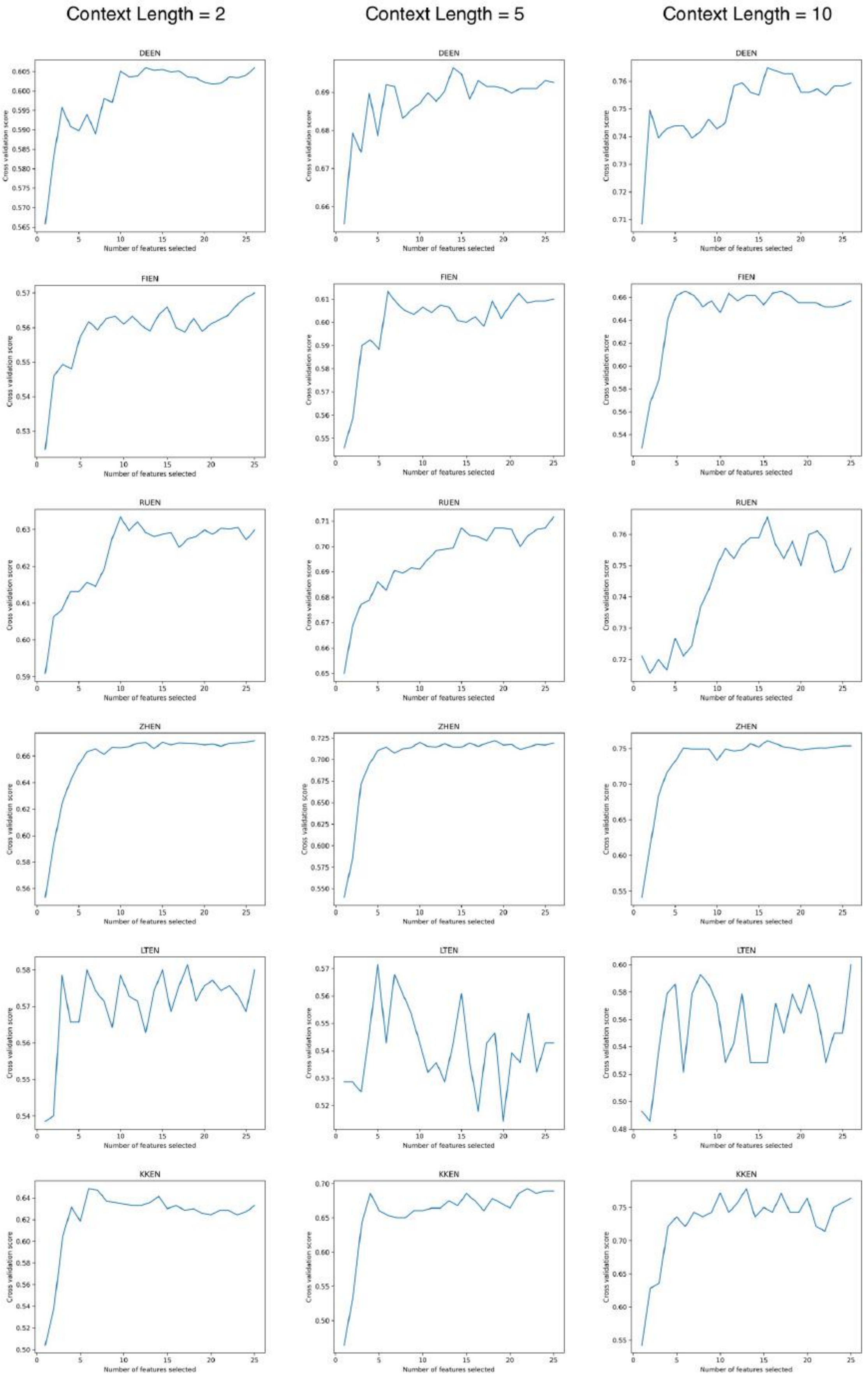


Figure 33: Cross-validation score with different number of selected features on training set during RFE for MT-HT.

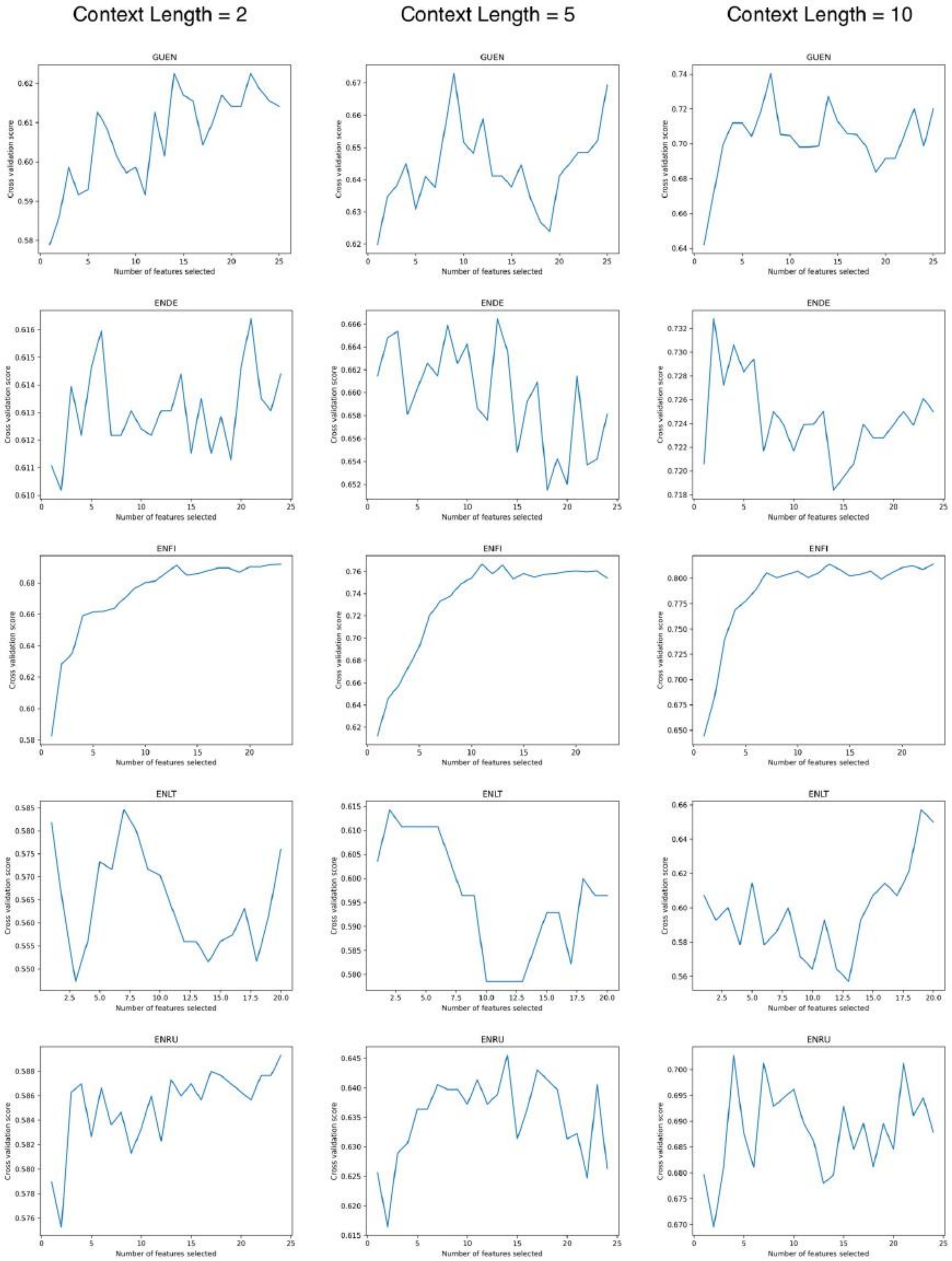


Figure 34: Cross-validation score with different number of selected features on training set during RFE for MT-HT.

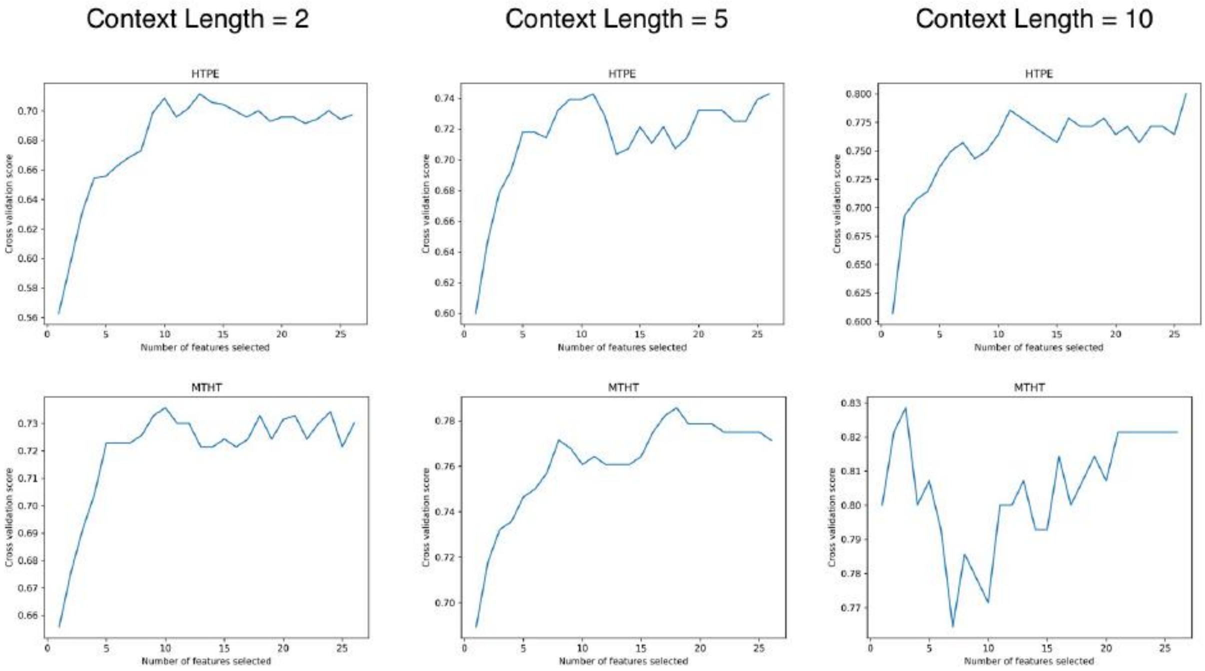


Figure 35: Cross-validation score with different number of selected features on training set during RFE for HT-PE.

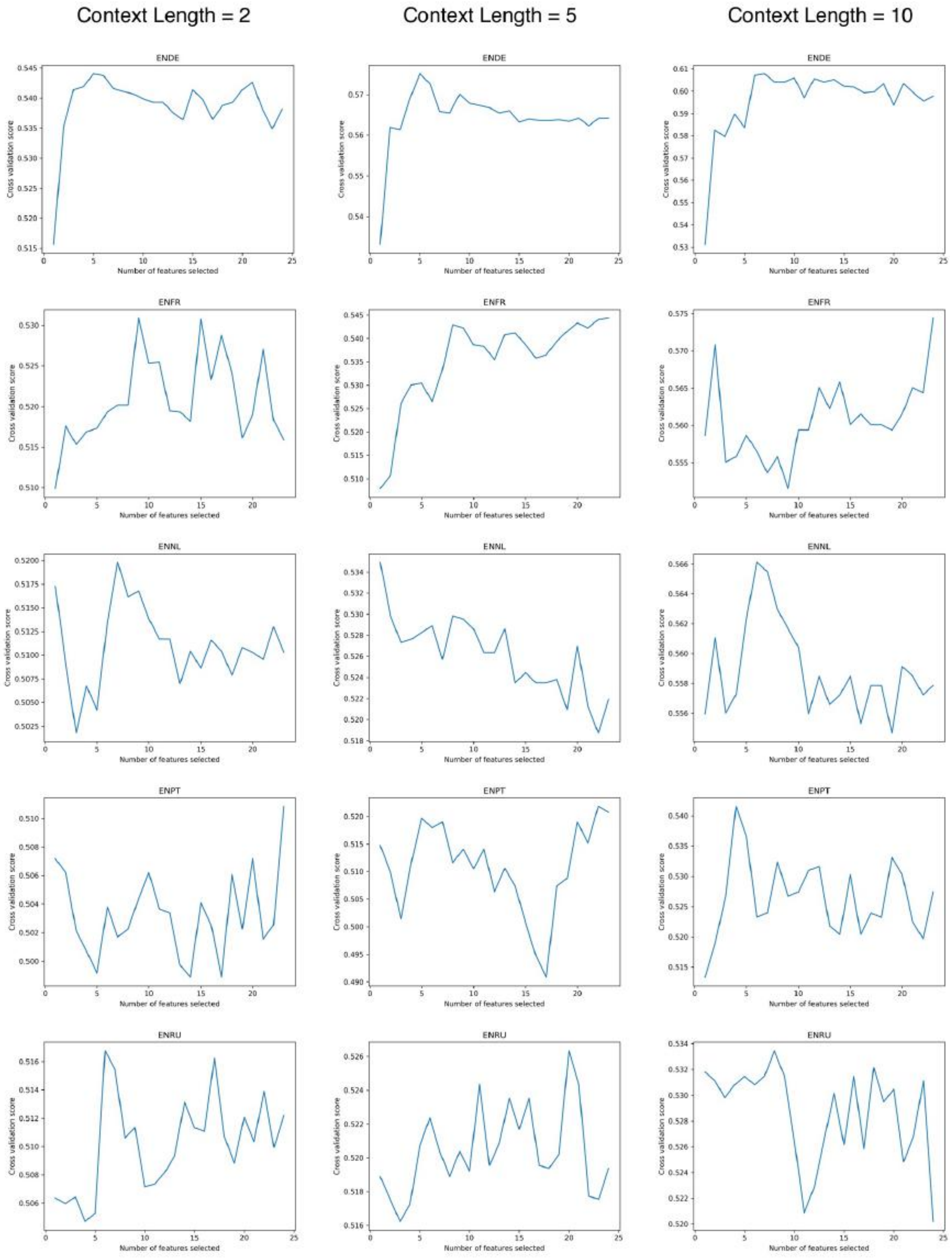


Figure 36: Cross-validation score with different number of selected features on training set during RFE for MT-PE.

E MT BY DEEPL VS. HT

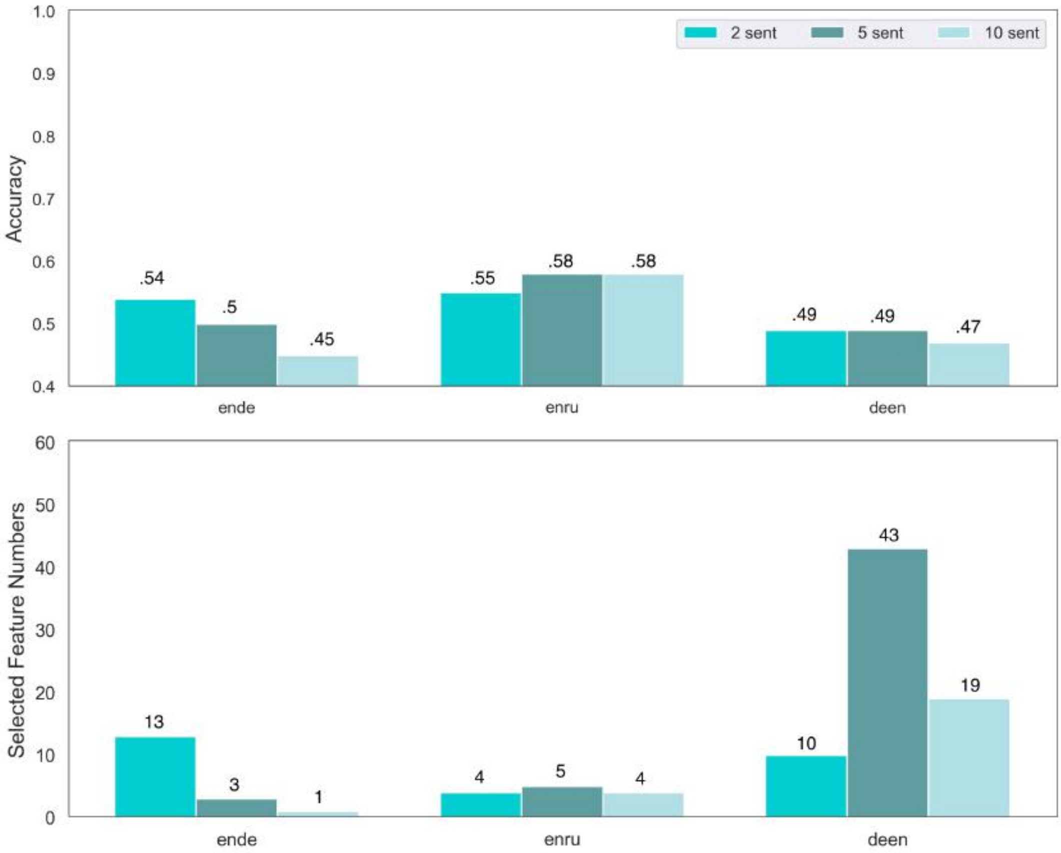


Figure 37: MT-HT classification results. Data is generated by DeepL and the classifier is trained with all features (including PoS and character ngrams.)

F ABSOLUTE COEFFICIENT RANKING

Feature name	Ave. rank	Std	Presence	Hypothesis
length ratio	2.8	2.95	85%	I
syllable ratio	3.9	3.38	55%	S
average sentence length	4.9	4.49	67%	S
yulesi	7.2	5.81	67%	S
mtld	7.2	6.52	52%	S
mean word rank	7.3	3.34	61%	S
5 most frequent word	7.4	5.08	70%	S
average word length	7.6	4.49	76%	S
pronoun ratio	8.3	6.08	67%	O
lexical density	8.8	5.44	58%	S
pmi	9.2	6.05	61%	N
10 most frequent word	9.5	4.73	52%	S
average function words	9.8	6.29	70%	E
thresold PMI	11.1	6.41	52%	N
type token ratio	11.3	7.17	48%	S
contextual function word	11.4	7.37	52%	I
repetition ratio	11.5	6.28	73%	N
50 most frequent word	11.6	5.54	52%	S
positional token frequency	12.3	5.11	48%	I
pos perplexity	12.3	5.23	39%	I
explicit naming	13.1	5.39	33%	E
passive verb ratio	13.5	5.75	48%	O
mdd	14.0	7.24	36%	S
perplexity	15.1	6.23	27%	S
mean multiple naming	15.2	3.97	33%	E
single naming	17.6	5.3	36%	E

Table 15: Feature ranking based on absolute coefficients of MT-HT classification. Translationese hypotheses are denoted as **S**implification, **N**ormalization, **E**xplicitation, **I**nterference, and **O**thers.

Feature name	Ave. rank	Std	Presence	Hypothesis
pmi	1.0		3%	N
length ratio	1.3	0.868	73%	I
50 most frequent word	2.0		3%	S
mean word rank	2.0		3%	S
mtld	2.4	1.673	15%	S
average function words	2.5	0.707	6%	E
syllable ratio	2.8	0.5	12%	S
mdd	3.3	1.799	21%	S
type token ratio	3.7	2.082	9%	S
repetition ratio	3.8	2.57	70%	N
pos perplexity	4.0	2.0	27%	I
passive verb ratio	4.1	2.265	48%	O
explicit naming	4.2	2.136	33%	E
yulesi	4.3	3.069	33%	S
perplexity	4.7	2.693	27%	S
contextual function word	4.9	3.643	24%	I
single naming	5.2	1.697	36%	E
mean multiple naming	5.5	2.423	33%	E
5 most frequent word	5.6	3.435	15%	S

Table 16: Feature (VIF < 5) ranking based on absolute coefficients of MT-HT classification.

Feature name	Ave. rank	Std	Presence	Hypothesis
length ratio	3.3	2.31	100%	I
positional token frequency	3.7	2.31	100%	I
50 most frequent word	5.0	1.73	100%	S
average function words	5.3	6.66	100%	E
mdd	5.5	3.54	67%	S
average word length	6.0	2.83	67%	S
mean word rank	6.7	3.51	100%	S
contextual function word	8.0	1.41	67%	I
mtld	10.0	12.73	67%	S
single naming	10.0	1.41	67%	E
average sentence length	10.0		33%	S
lexical density	10.7	3.51	100%	S
pmi	11.0	9.9	67%	N
pronoun ratio	11.0		33%	O
passive verb ratio	12.0		33%	O
repetition ratio	13.0		33%	N
thresold PMI	13.3	11.15	100%	N
type token ratio	14.0	14.14	67%	S
perplexity	15.0	9.9	67%	S
mean multiple naming	15.0		33%	E
yulesi	15.0	7.07	67%	S
syllable ratio	16.0		33%	S
10 most frequent word	17.0		33%	S
pos perplexity	17.5	7.78	67%	I
5 most frequent word	21.0		33%	S
explicit naming	25.0		33%	E

Table 17: Feature ranking based on absolute coefficients of HT-PE classification. Translationese hypotheses are denoted as **S**implification, **N**ormalization, **E**xplicitation, **I**nterference, and **O**thers.

Feature name	Ave. rank	Std	Presence	Hypothesis
mdd	1.0	0.0	67%	S
contextual function word	1.0		33%	I
single naming	2.0		33%	E
yulesi	2.0		33%	S
pos perplexity	2.5	0.71	67%	I
passive verb ratio	3.0		33%	O
mean multiple naming	4.0		33%	E
perplexity	4.5	2.12	67%	S
explicit naming	5.0		33%	E

Table 18: Feature (VIF < 5) ranking based on absolute coefficients of HT-PE classification.

Feature name	Ave. rank	Std	Presence	Hypothesis
length ratio	4.4	5.68	93%	I
average sentence length	6.0	3.37	27%	S
pmi	6.1	6.47	53%	N
50 most frequent word	6.4	3.7	53%	S
thresold PMI	6.8	4.64	67%	N
mtld	7.4	8.0	60%	S
yulesi	7.7	4.15	47%	S
lexical density	7.9	6.85	53%	S
type token ratio	8.0	4.8	47%	S
repetition ratio	8.1	6.17	60%	N
mean multiple naming	8.3	8.02	60%	E
average function words	8.5	7.84	73%	E
pos perplexity	8.5	3.54	13%	I
5 most frequent word	9.0	7.75	27%	S
explicit naming	9.1	6.56	53%	E
positional token frequency	10.0	6.4	33%	I
pronoun ratio	10.5	5.28	40%	O
single naming	11.0	9.55	40%	E
mean word rank	11.5	6.86	40%	S
10 most frequent word	13.2	5.78	40%	S
average word length	13.5	3.83	40%	S
mdd	14.2	6.18	33%	S
passive verb ratio	14.8	7.57	40%	O
contextual function word	16.4	1.52	33%	I

Table 19: Feature ranking based on absolute coefficients of MT-PE classification. Translationese hypotheses are denoted as **S**implification, **N**ormalization, **E**xplicitation, **I**nterference, and **O**thers.

Feature name	Ave. rank	Std	Presence	Hypothesis
average function words	1.5	0.71	13%	E
mtld	1.8	0.5	27%	S
pmi	2.0		7%	N
mean multiple naming	2.0	1.5	60%	E
single naming	2.7	0.52	40%	E
yulesi	3.0	2.65	20%	S
repetition ratio	3.0	2.2	53%	N
length ratio	3.1	2.12	73%	I
pos perplexity	3.5	2.12	13%	I
type token ratio	4.3	1.53	20%	S
explicit naming	5.0	2.0	53%	E
50 most frequent word	5.0		7%	S
passive verb ratio	5.5	2.74	40%	O
mdd	5.6	1.82	33%	S
10 most frequent word	6.5	2.12	13%	S
positional token frequency	11.0		7%	I
contextual function word	12.0		7%	I

Table 20: Feature (VIF < 5) ranking based on absolute coefficients of MT-PE classification.

G AVERAGE COEFFICIENTS RANKING

Feature name	Ave. coef	Std	Presence	Hypothesis
syllable ratio	-0.459	1.318	55%	S
mtld	-0.389	0.414	52%	S
length ratio	-0.364	0.426	85%	I
pronoun ratio	-0.257	0.225	67%	O
lexical density	-0.237	0.31	58%	S
repetition ratio	-0.195	0.185	73%	N
mean word rank	-0.193	0.188	61%	S
pmi	-0.186	0.302	61%	N
mdd	-0.136	0.156	36%	S
mean multiple naming	-0.109	0.215	33%	E
50 most frequent word	-0.061	0.223	52%	S
pos perplexity	-0.05	0.178	39%	I
single naming	-0.041	0.121	36%	E
10 most frequent word	-0.037	0.322	52%	S
type token ratio	-0.015	0.507	48%	S
average function words	0.008	0.273	70%	E
average word length	0.022	0.305	76%	S
passive verb ratio	0.038	0.228	48%	O
contextual function word	0.04	0.368	52%	I
explicit naming	0.055	0.189	33%	E
positional token frequency	0.08	0.252	48%	I
thresold PMI	0.097	0.345	52%	N
yulesi	0.173	0.449	67%	S
5 most frequent word	0.292	0.27	70%	S
average sentence length	0.345	1.116	67%	S
perplexity	1.171	3.298	27%	S

Table 21: Average coefficients of MT-HT classification. Average coefficient is calculated across 11 language pairs and context length of 2, 5, and 10.

Feature name	Ave. coef	Std	Presence	Hypothesis
mtld	-0.483	0.487	15%	S
length ratio	-0.32	0.453	73%	I
yulesi	-0.216	0.352	33%	S
pmi	-0.213		3%	N
repetition ratio	-0.206	0.209	70%	N
mdd	-0.146	0.196	21%	S
syllable ratio	-0.118	0.086	12%	S
mean word rank	-0.076		3%	S
mean multiple naming	-0.061	0.099	33%	E
pos perplexity	-0.047	0.105	27%	I
single naming	-0.032	0.113	36%	E
explicit naming	-0.027	0.189	33%	E
type token ratio	0.029	0.014	9%	S
passive verb ratio	0.064	0.199	48%	O
perplexity	0.079	0.125	27%	S
average function words	0.173	0.149	6%	E
contextual function word	0.175	0.221	24%	I
5 most frequent word	0.179	0.337	15%	S
50 most frequent word	0.276		3%	S

Table 22: Average coefficients of MT-HT classification with features whose VIF < 5.

Feature name	Ave. coef	Std	Presence	Hypothesis
average function words	-0.645	0.545	100%	E
pronoun ratio	-0.612		33%	O
length ratio	-0.581	0.396	100%	I
passive verb ratio	-0.561		33%	O
50 most frequent word	-0.55	0.341	100%	S
repetition ratio	-0.545		33%	N
mean multiple naming	-0.519		33%	E
pmi	-0.513	0.131	67%	N
average word length	-0.484	0.494	67%	S
mdd	-0.479	0.418	67%	S
mean word rank	-0.461	0.441	100%	S
single naming	-0.451	0.292	67%	E
10 most frequent word	-0.431		33%	S
lexical density	-0.356	0.228	100%	S
mtld	-0.323	0.111	67%	S
thresold PMI	-0.242	0.31	100%	N
type token ratio	-0.129	0.052	67%	S
yulesi	-0.098	0.285	67%	S
pos perplexity	-0.096	0.015	67%	I
explicit naming	-0.05		33%	E
5 most frequent word	0.223		33%	S
perplexity	0.284	0.123	67%	S
syllable ratio	0.441		33%	S
contextual function word	0.53	0.361	67%	I
positional token frequency	0.622	0.398	100%	I
average sentence length	0.633		33%	S

Table 23: Average coefficients HT-PE classification.

Feature name	Ave. coef	Std	Presence	Hypothesis
mdd	-0.498	0.426	67%	S
contextual function word	-0.44		33%	I
passive verb ratio	-0.239		33%	O
pos perplexity	-0.163	0.134	67%	I
mean multiple naming	-0.142		33%	E
single naming	-0.137		33%	E
yulesi	-0.109		33%	S
explicit naming	0.015		33%	E
perplexity	0.055	0.086	67%	S

Table 24: Average coefficients of HT-PE classification with features whose values are < 5.

Feature name	Ave. coef	Std	Presence	Hypothesis
50 most frequent word	-0.117	0.191	53%	S
repetition ratio	-0.114	0.136	60%	N
thresold PMI	-0.104	0.101	67%	N
passive verb ratio	-0.069	0.11	40%	O
yulesi	-0.054	0.189	47%	S
type token ratio	-0.054	0.212	47%	S
average sentence length	-0.053	0.201	27%	S
lexical density	-0.033	0.091	53%	S
contextual function word	-0.029	0.064	33%	I
mean word rank	-0.026	0.1	40%	S
average word length	-0.018	0.075	40%	S
explicit naming	-0.016	0.108	53%	E
10 most frequent word	-0.012	0.113	40%	S
mdd	-0.004	0.025	33%	S
single naming	0.034	0.034	40%	E
5 most frequent word	0.035	0.139	27%	S
pronoun ratio	0.046	0.045	40%	O
average function words	0.066	0.118	73%	E
positional token frequency	0.067	0.083	33%	I
mtld	0.094	0.108	60%	S
pos perplexity	0.135	0.017	13%	I
mean multiple naming	0.156	0.198	60%	E
length ratio	0.203	0.237	93%	I
pmi	0.214	0.22	53%	N

Table 25: Average coefficients of MT-PE classification. Average coefficient is calculated across 4 language pairs and context length of 2, 5, and 10.

Feature name	Ave. coef	Std	Presence	Hypothesis
50 most frequent word	-0.225		7%	S
repetition ratio	-0.151	0.194	53%	N
passive verb ratio	-0.062	0.102	40%	O
10 most frequent word	-0.047	0.054	13%	S
yulesi	-0.041	0.074	20%	S
contextual function word	-0.009		7%	I
explicit naming	0.014	0.058	53%	E
mdd	0.015	0.014	33%	S
positional token frequency	0.036		7%	I
single naming	0.061	0.081	40%	E
type token ratio	0.111	0.128	20%	S
average function words	0.118	0.242	13%	E
mtld	0.121	0.123	27%	S
pos perplexity	0.16	0.042	13%	I
length ratio	0.161	0.226	73%	I
mean multiple naming	0.205	0.23	60%	E
pmi	0.333		7%	N

Table 26: Average coefficients of MT-PE classification with features whose values are < 5.

H CORRELATION PLOT

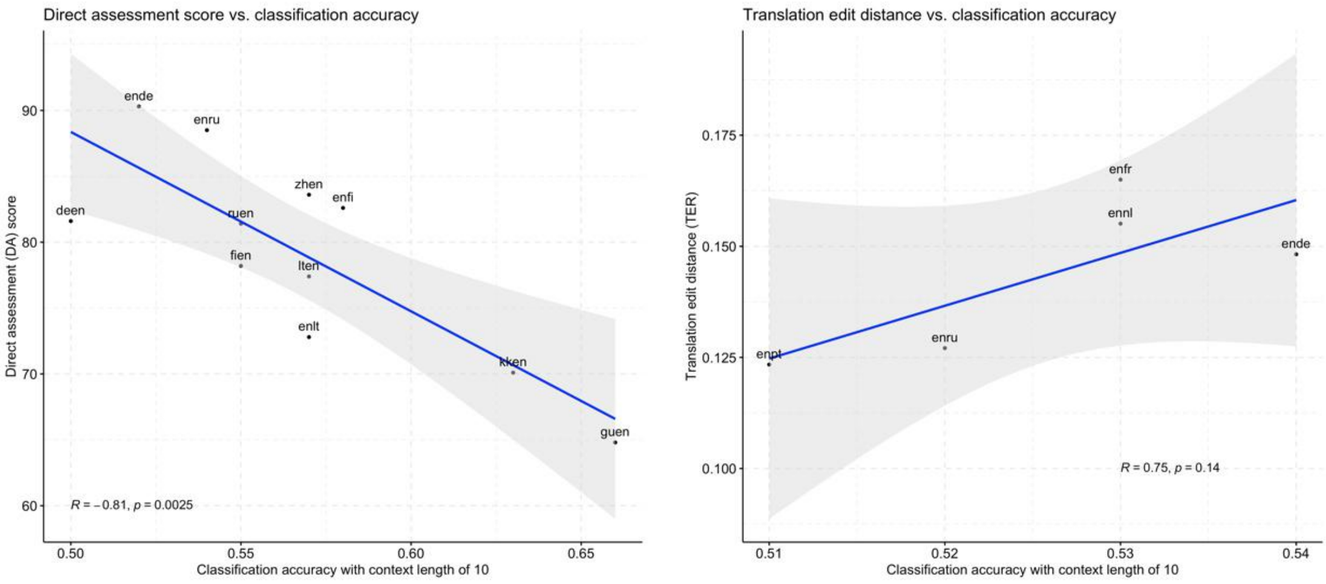


Figure 38: Pearson correlation between and classification accuracy and the direct assessment score (left) and translation edit rate (right). The accuracy is obtained with context length of 2.

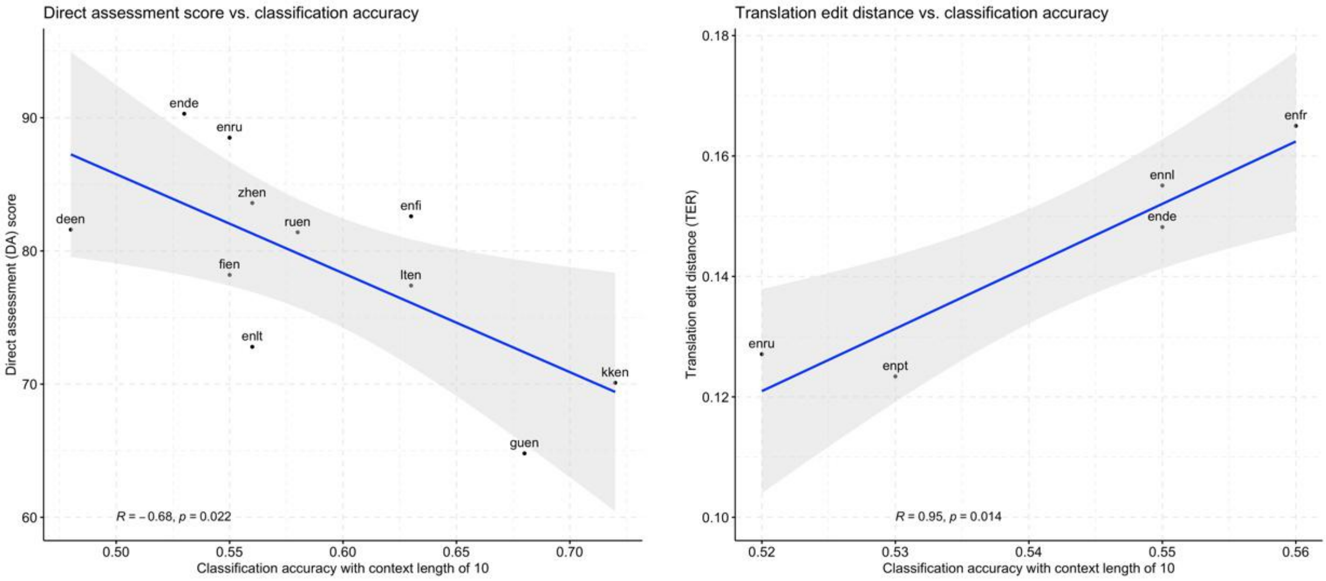


Figure 39: Pearson correlation between and classification accuracy and the direct assessment score (left) and translation edit rate (right). The accuracy is obtained with context length of 5.

BIBLIOGRAPHY

Roee Aharoni, Moshe Koppel, and Yoav Goldberg. 2014. Automatic detection of machine translated text and translation quality estimation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 289–295, Baltimore, Maryland. Association for Computational Linguistics.

Lars Ahrenberg. 2017. Comparing machine translation and human translation: A case study. pages 21–28.

Yuki Arase and Ming Zhou. 2013. Machine translation detection from monolingual web-text. volume 1, pages 1597–1607.

Eleftherios Avramidis, Aljoscha Burchardt, Sabine Hunsicker, Maja Popović, Cindy Tscherwinka, David Vilar, and Hans Uszkoreit. 2014. The taraXÜ corpus of human-annotated machine translations. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2679–2682, Reykjavik, Iceland. European Language Resources Association (ELRA).

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural machine translation by jointly learning to align and translate.

M. Baker. 1995. 'corpora in translation studies: An overview and some suggestions for future research'. *Targets*, 7:223–243.

Mona Baker, Gill Francis, and Elena Tognini-Bonelli. 1993. *'Corpus Linguistics and Translation Studies: Implications and Applications'*. John Benjamins Publishing Company, Netherlands.

Mona Baker and Harold Somers. 1996. *'Corpus-based Translation Studies: The Challenges that Lie Ahead'*. John Benjamins Publishing Company, Netherlands.

Mohit Bansal, Chris Quirk, and Robert Moore. 2011. Gappy phrasal alignment by agreement. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1308–1317, Portland, Oregon, USA. Association for Computational Linguistics.

Marco Baroni and Silvia Bernardini. 2005. A New Approach to the Study of Translationese: Machine-learning the Difference between Original and Translated Text. *Literary and Linguistic Computing*, 21(3):259–274.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Nitsa Ben-Ari. 1998. The ambivalent case of repetitions in literary translation. avoiding repetitions: a 'universal' of translation? *Meta*, 43(1):68–78.

Kenneth Benoit, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. 2018. quanteda: An r package for the quantitative analysis of textual data. *Journal of Open Source Software*, 3:774.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study.

Alexandra Birch and Miles Osborne. 2011. Reordering metrics for mt. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, page 1027–1035, USA. Association for Computational Linguistics.

Christopher M. Bishop. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

Shoshana Blum and E. A. Levenston. 1978. Universals of lexical simplification. *Language Learning*, 28(2):399–415.

S. Blum-kulka and J. House. 1996. Shifts of cohesion and coherence in translation.

Kristine Bundgaard. 2017. Translator attitudes towards translator-computer interaction - findings from a workplace study. *HERMES - Journal of Language and Communication in Business*, (56):125–144.

Brandon Butcher and Brian Smith. 2020. Feature engineering and selection: A practical approach for predictive models. *The American Statistician*, 74:308–309.

Dave Carter and Diana Inkpen. 2012. Searching for poor quality machine translated text: Learning the difference between human writing and machine translations. In *Canadian Conference on AI*.

M. Cettolo, Niehues Jan, Stüker Sebastian, L. Bentivogli, R. Cattoni, and Marcello Federico. 2016. The iwslt 2016 evaluation campaign.

Jieun Chae and Ani Nenkova. 2009. Predicting the fluency of text with shallow structural features: Case studies of machine translation and human-written text. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 139–147, Athens, Greece. Association for Computational Linguistics.

Rajen Chatterjee, Christian Federmann, Matteo Negri, and Marco Turchi. 2019. Findings of the WMT 2019 shared task on automatic post-editing. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 11–28, Florence, Italy. Association for Computational Linguistics.

Oliver Čulo and Jean Nitzke. 2016. Patterns of terminological variation in post-editing and of cognate use in machine translation in contrast to human translation. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 106–114.

Joke Daems, Orphée De Clercq, and Lieve Macken. 2017. Translationese and post-editese: How comparable is comparable quality? *Linguistica Antverpiensia, New Series – Themes in Translation Studies*, 16:89–103.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Yadolah Dodge. 2008. *The Concise Encyclopedia of Statistics*. Springer-Verlag New York.

Michael Farrell. 2018. Machine translation markers in post-edited machine translation output. In *Proceedings of the 40th Conference Translating and the Computer*, pages 50–59.

William Frawley. 1984. Prolegomenon to a theory of translation. In William Frawley, editor, *Translation: Literary, Linguistic and Philosophical Perspectives*, page 159–175. Associated University Press.

Ignacio Garcia. 2010. Is machine translation ready yet? *Target*, 22:7–21.

Martin Gellerstam. 1986. Translationese in swedish novels translated from english.

Spence Green, Jeffrey Heer, and Christopher D. Manning. 2013. The efficacy of human post-editing for language translation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, page 439–448, New York, NY, USA. Association for Computing Machinery.

Ana Guerberof Arenas. 2009. Productivity and quality in mt post-editing.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422.

Hassan, Hany Awadalla, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, Will Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. 2018. Achieving human parity on automatic chinese to english news translation. ArXiv:1803.05567.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Chih-wei Hsu, Chih-chung Chang, and Chih-Jen Lin. 2003. A practical guide to support vector classification chih-wei hsu, chih-chung chang, and chih-jen lin.

Ke Hu and Patrick Cadwell. 2016. A comparative study of post-editing guidelines. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 34206–353.

Richard Hudson. 1995. Measuring syntactic difficulty. *Manuscript, University College, London.*

Iustina Ilisei and Diana Inkpen. 2011. Translationese traits in romanian newspapers: A machine learning approach. *International Journal of Computational Linguistics and Applications*, 2.

Iustina Ilisei, Diana Inkpen, Gloria Corpas Pastor, and Ruslan Mitkov. 2010. Identification of translationese: A machine learning approach. volume 6008, pages 503–511.

J. Ive, Lucia Specia, Sara Szoc, Tom Vanallemeersch, J. V. D. Bogaert, E. Farah, Christine Maroti, Artur Ventura, and M. Khalilov. 2020. A post-editing dataset in the legal domain: Do we underestimate neural machine translation quality? In *LREC*.

Daniel Jurafsky and James H. Martin. 2009. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., USA.

D. Kenny. 2001. Lexis and creativity in translation : a corpus-based study.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation.

Maarit Koponen. 2016. Is machine translation post-editing worth the effort? a survey of research into post-editing and effort. *The Journal of Specialised Translation*, pages 131–148.

Moshe Koppel and Noam Ordan. 2011. Translationese and its dialects. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1318–1326, Portland, Oregon, USA. Association for Computational Linguistics.

Maria Kunilovskaya and Andrey Kutuzov. 2017. Universal dependencies-based syntactic features in detecting human translation varieties. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 27–36, Prague, Czech Republic.

Samuel Läubli, Rico Sennrich, and Martin Volk. 2018. Has machine translation achieved human parity? a case for document-level evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4791–4796, Brussels, Belgium. Association for Computational Linguistics.

Sara Laviosa. 1998. Core patterns of lexical use in a comparable corpus of english narrative prose. *Meta: Journal des traducteurs*, 43.

Sara Laviosa. 2002. *Corpus-based Translation Studies: Theory, Findings, Applications*. Brill | Rodopi, Leiden, The Netherlands.

Gennadi Lembersky, Noam Ordan, and Shuly Wintner. 2011. Language models for machine translation: Original vs. translated texts. In *Proceedings of the 2011*

Conference on Empirical Methods in Natural Language Processing, pages 363–374, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Haitao Liu. 2008. Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science*, 9:159–191.

Adam Lopez. 2008. Statistical machine translation. *ACM Comput. Surv.*, 40(3).

J. A. Martín and Anna Civil Serra. 2014. Integration of a machine translation system into the editorial process flow of a daily newspaper. *Proces. del Leng. Natural*, 53:193–196.

Philip Mccarthy and Scott Jarvis. 2010. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods*, 42:381–92.

Philip M McCarthy. 2005. An assessment of the range and usefulness of lexical diversity measures and the potential of the measure of textual, lexical diversity (mtld). *PhD Thesis, Dissertation Abstracts International*, 66:12.

Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining. 2012. *Introduction to Linear Regression Analysis*. Wiley-Interscience.

Joss Moorkens, Antonio Toral, Sheila Castilho, and Andy Way. 2018. Translators' perceptions of literary post-editing using statistical and neural machine translation. *Translation Spaces*, 7(2):240–262.

Matteo Negri, Marco Turchi, Rajen Chatterjee, and Nicola Bertoldi. 2018. ESCAPE: a large-scale synthetic corpus for automatic post-editing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Hoang-Quoc Nguyen-Son, Tran Phuong Thao, Seira Hidano, and Shinsaku Kiyomoto. 2019. Detecting machine-translated paragraphs by matching similar words.

Michael Oakes and Meng Ji. 2012. *Quantitative Methods in Corpus-Based Translation Studies: A practical guide to descriptive translation research*.

Sharon O'Brien, Laura Winther Balling, Michael Carl, Michel Simard, and Lucia Specia, editors. 2014. *Post-editing of Machine Translation: Processes and Applications*. Cambridge Scholars Publishing, United Kingdom.

Maeve Olohan and Mona Baker. 2000. Reporting that in translated english. evidence for subconscious processes of explicitation? *Across Languages and Cultures*, 1:141–158.

L. Øverås. 1998. In search of the third code: An investigation of norms in literary translation. *Meta: Translators' Journal*, 43:557–570.

James Pennebaker, M. Francis, and R. Booth. 2001. Linguistic inquiry and word count (liwc): Liwc2001. 71.

Mirko Plitt and François Masselot. 2010. A productivity test of statistical machine translation post-editing in a typical localisation context. *The Prague Bulletin of Mathematical Linguistics*, 93:7–16.

Maja Popovic. 2020. On the differences between human translations. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 365–374, Lisboa, Portugal. European Association for Machine Translation.

Larry Selinker. 1972. Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1-4):209 – 232.

Omar Sheikh al Shabab. 1996. Interpretation and the language of translation : creativity and conventions in translation.

Matthew Snover, Bonnie J. Dorr, R. Schwartz, and L. Micciulla. 2006. A study of translation edit rate with targeted human annotation.

Robyn Speer, Joshua Chin, Andrew Lin, Sara Jewett, and Lance Nathan. 2018. Luminosoinsight/wordfreq: v2.2.

Milan Straka. 2018. UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.

TAUS. 2016. *TAUS Post-Editing Guidelines*.

Elke Teich. 2003. *Cross-Linguistic Variation in System and Text: A Methodology for the Investigation of Translations and Comparable Texts*. Walter de Gruyter Berlin.

Antonio Toral. 2019. Post-editese: an exacerbated translationese. *CoRR*, abs/1907.00900.

Antonio Toral. 2020. Reassessing claims of human parity and super-human performance in machine translation at wmt 2019.

Antonio Toral, Sheila Castilho, Ke Hu, and Andy Way. 2018a. Attaining the unattainable? reassessing claims of human parity in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 113–123, Brussels, Belgium. Association for Computational Linguistics.

Antonio Toral and Víctor M. Sánchez-Cartagena. 2017. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1063–1073, Valencia, Spain. Association for Computational Linguistics.

Antonio Toral, Martijn Wieling, and Andy Way. 2018b. Post-editing effort of a novel with statistical and neural machine translation. *Frontiers in Digital Humanities*, 5:9.

G. Toury. 1979. Interlanguage and its manifestations in translation. *Meta: Translators' Journal*, 24:223–231.

G Toury. 1995. *Descriptive Translation Studies - and beyond*. Amsterdam/Philadelphia: John Benjamins Pub. Co.

Eva Vanmassenhove, Dimitar Shterionov, and Andy Way. 2019. Lost in translation: Loss and decay of linguistic richness in machine translation. *CoRR*, abs/1906.12068.

Vered Volansky, Noam Ordan, and Shuly Wintner. 2013. On the features of translationese. *Digital Scholarship in the Humanities*, 30(1):98–118.

G. Udny Yule. 1944. *The Statistical Study of Literary Vocabulary*. Cambridge University Press, Cambridge, USA.

Mike Zhang and Antonio Toral. 2019. The effect of translationese in machine translation test sets. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 73–81, Florence, Italy. Association for Computational Linguistics.

A Zisserman. 2015. Lecture 2: The svm classifier. *C19 Machine Learning lectures Hilary 2015*. `http://www.robots.ox.ac.uk/~az/lectures/ml/lect2.pdf`, last accessed on 29-11-2020.