

On the Influence of Machine Translation on Language Origin Obfuscation

Benjamin Murauer, Michael Tschuggnall, and Günther Specht
{firstname.lastname}@uibk.ac.at

Universität Innsbruck

Abstract. In the last decade, machine translation has become a popular means to deal with multilingual digital content. By providing higher quality translations, obfuscating the source language of a text becomes more attractive. In this paper, we analyze the ability to detect the source language from the translated output of two widely used commercial machine translation systems by utilizing machine-learning algorithms with basic textual features like n-grams. Evaluations show that the source language can be reconstructed with high accuracy for documents that contain a sufficient amount of translated text. In addition, we analyze how the document size influences the performance of the prediction, as well as how limiting the set of possible source languages improves the classification accuracy.

1 Introduction

The amount of textual data being shared through various channels over the Internet is growing constantly. Considering the fact that this data is transferred across cultural as well as political borders, machine-translation is an often used means in case the source or target language is understood insufficiently. On the other hand, the growing quality of publicly available services from, e.g., Google or Microsoft, can also be used to obfuscate the source language, and thus the geographical origin. While there exist many approaches in the field of stylometry to identify the author of documents [17] (which is also frequently used in digital forensics [13]), to determine the native language of a person based on samples written in a non-native language (Native Language Identification, e.g., [8]), or to detect the translation tool of a machine-translated text [1] it has not been investigated whether similar techniques can be utilized to predict the source language of machine-translated text with satisfying accuracy. Algorithms which are able to reconstruct the original language could be utilized for, e.g., detecting plagiarism: once a part of a document is confirmed to be machine-translated and the original language is known, further investigations can focus on that languages. Regarding forensics, a use case could include determining the native language of a threat that has been automatically translated, e.g., to restrict the geographical origin and consequently the amount of possible authors.

We analyze the prediction performance of commonly used stylometric features such as n-grams – which are known to work well for various text classification problems [8, 12] – regarding the detection of the source language from machine-translated text. Adhering to the fact that most of online text is shared in English [14], we restrict the target language to be English as well for this study.

Analogous to the field of authorship attribution where it has been shown that an increasing amount of possible authors decreases the prediction performance, we additionally analyze the influence of the number of possible source languages on the quality of the prediction. Finally, we determine the amount of text that is required to achieve a specific accuracy for predicting the source language.

Concretely, we answer two main research questions (RQ) in this paper:

RQ1: Given a document, can it be detected whether it was machine-translated, and if yes, how accurate can the source language be determined?

RQ2: How do different feature sets, varying amounts of possible source languages and document lengths influence the prediction accuracy?

We find that using traditional features, the original source language of a translated text can be predicted with a very high accuracy. Additionally, we show that this prediction is possible with even very small amounts of text for many possible languages.

2 Related Work

Previous work in authorship attribution [5, 16] has shown that the use of automatic translation techniques actually adds information to a text document since the translation itself adds details specific to the methods used. We assume this fact is still valid although the engine used by the translation services has changed from statistical methods to neural networks^{1,2}.

A field of research closely related to the problem presented in this paper is Native Language Identification (NLI), which aims to find the native language of authors given only documents in a foreign language. Features used for this classification task range from traditional n-grams [11, 2] over string kernels [7] to detailed stylistic analyses of the words and grammar that was used by the authors [4, 3, 18, 19].

In an approach more closely related to the problem addressed by this paper, Koppel et al. determine the source language of human-translated documents by analyzing various types of errors introduced by non-native authors [10].

The effect of automatic machine translation on text has been analyzed with focus on different aspects. Detecting the tools which have been used to translate a text has been shown to be already possible for very small amounts of text [1]. Caliskan and Greenstadt show that repeated translation of text does not obfuscate an author's individual writing style [5]. Instead, the translation itself adds even more information to the text, making it possible to determine which translation engine was used. In their research, a text that has been translated two and three times is shown to still contain enough information to determine the authorship with an accuracy of over 90%. Similarly, it has been shown that the effectiveness of using machine-translation to obfuscate the original text and thus reduce the accuracy of authorship verification is limited [15].

¹ <https://www.blog.google/products/translate/found-translation-more-accurate-fluent-sentences-google-translate/>

² <https://blogs.msdn.microsoft.com/translation/2016/11/15/microsoft-translator-launching-neural-network-based-translations-for-all-its-speech-languages/>

3 Methodology

At a glance, we utilize textual features commonly used in stylometry to quantify the characteristics of translated English text originating from a specific source language. These features are then used to train state-of-the-art machine-learning classifiers, which are finally used to predict the source language.

3.1 Dataset

We used two different sources to construct a representative dataset containing German (de), English (en), Spanish (es), French (fr), Italian (it), Japanese (ja), Korean (ko), Dutch (nl), Turkish (tr) and Chinese (zh) text: firstly, the Leipzig Corpus Collection (further called *lcc*) [6] contains random sentences from news articles, web pages and Wikipedia articles. Secondly, random Wikipedia articles were selected for each of the languages mentioned above, which will collectively be referred to as *wiki*.

For each language from both *lcc* and *wiki*, 5,000 sentences were chosen randomly and translated to English using the services of Google³ and Microsoft⁴. Subsequently, 9 of the 10 languages are translated, whereas the original English sentences are not translated. The dataset generation process is displayed on Fig. 1, where in the first step the sentences from both datasets are combined, yielding 10,000 sentences for each language.

Thereby, the translation itself was performed sentence-wise. Both translation engines feature an automatic detection of the source language, however, since this information was known it was provided to them. This implicitly enables our prediction model to detect whether a document has been machine-translated at all. For all further investigations, the texts from the two services were kept separate, i.e., the computed models were tested on documents coming from the same translation service that was also used for training. While the performance of cross-translation-service evaluations might be of interest at a first glance, this was omitted since it has already been shown that detecting the tool which was used to translate a text can be detected easily [1], i.e., in a realistic scenario one would probably use such an approach as a preliminary step to choose the correct model.

Document Sampling To answer parts of RQ2, i.e., the influence of text length on the source language prediction quality, documents of various text sizes are required. To construct expressive documents, the translated sentences were combined randomly to create documents of various sizes, i.e., data sets consisting of short (S), medium (M), long (L) and very long (XL) documents, respectively⁵. The outcome of this step of the workflow is depicted in the bottom part of Fig. 1, and details about the resulting datasets are listed in Table 1.

³ <https://translate.google.com>

⁴ <https://www.microsoft.com/en-us/translator/>

⁵ The sampling on a random basis was required since the *lcc* dataset only contains random sentences, rather than cohesive documents.

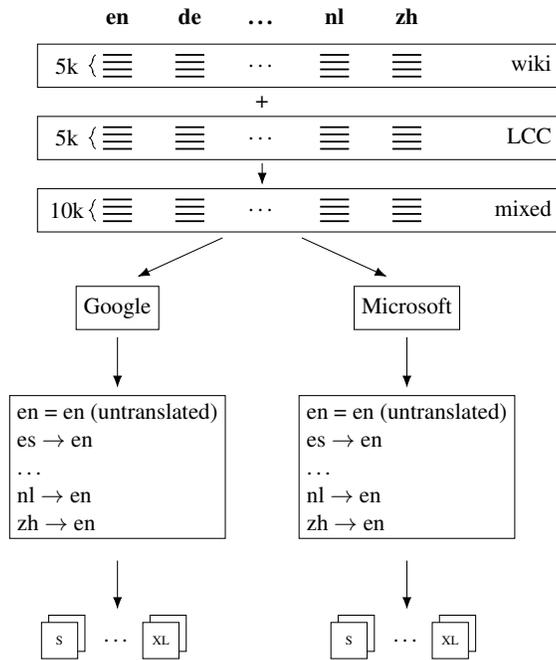


Fig. 1. Dataset Generation Workflow

The choice for the document sizes is thereby loosely based on typical sizes of on-line texts according to social media statistics⁶. All documents created are unique with respect to their sentences contained, i.e., one sentence can only be part of one document per size class. This limitation causes the data set with the longest documents (XL) to contain less documents than the other datasets, for which a maximum of 1,000 documents were created.

Table 1. Data Sets with Varying Document Lengths

length	no. of words	no. of docs
short (S)	5–50	1,000
medium (M)	51–200	1,000
long (L)	201–1,000	1,000
xlong (XL)	≥ 1,001	312 (on avg)

⁶ e.g., <https://blog.bufferapp.com/optimal-length-social-media>

3.2 Features

To distinguish the origin of the translated documents, several text features are used, which are listed in Table 2. These features were calculated by the Python library *scikit-learn*⁷ as well as the language processing library *spaCy*⁸ for POS tagging. Regarding the features used, the punctuation occurrence counts the amount of selected punctuation marks⁹ in each document. Features 2 and 3 count the occurrences of distinct words, separated by whitespace, whereby feature 3 disregards all English stop words¹⁰. Features 4-23 count the occurrences of n-grams of various lengths of characters and POS-tags, respectively. The n-gram-features denote the combination of all n-grams ranging from length 1 to 4. The *tf-idf* term denotes that the respective measure is normalized using term frequency-inverse document frequency metric provided by *scikit-learn*¹¹.

Combinations of features were tested, but provided no additional accuracy over using single features. This is displayed in Section 4.

Table 2. List of Utilized Features

id	feature name
1	punctuation occurrence
2	word occurrence
3	word occurrence, w/o stop words
4-8	character 1, 2, 3, 4, n-grams
9-13	character 1, 2, 3, 4, n-grams w. tf-idf
14-18	POS-tag 1, 2, 3, 4, n-grams
19-23	POS-tag 1, 2, 3, 4, n-grams w. tf-idf

3.3 Classifier

In order to determine the classifier for this task, a preliminary evaluation has been conducted on the S-dataset, using all features described in Table 2 individually. Thereby, several popular classifiers were tested: Naive Bayes, K Nearest Neighbors (KNN), Support Vector Machines (SVM) with Linear and RBF Kernels, Random Forest and a Multilayer Perceptron (MLP). This preliminary step was required due to the infeasibility of testing all classifiers with all experimental runs described in Section 4. To estimate the classifier performances, the default parameters given by the *scikit-learn* library have been used. The results of this preliminary run are displayed in Table 3. We observe that the linear SVM outperformed all other classifiers, if only by a small margin. The neural network (MLP) has a similar performance, at the cost of a much longer runtime. Being

⁷ <http://scikit-learn.org/>

⁸ <https://spacy.io/>

⁹ ; , . ? ! " # \$ % & () [] { } * + - \ / ^ _ ` ' ~

¹⁰ We rely on the list of stop words provided by *scikit-learn*

¹¹ http://scikit-learn.org/stable/modules/feature_extraction.html#tfidf-term-weighting

Table 3. Accuracy of Different Classifiers on the Short Data Set. The feature ids correspond to the features listed in Table 2. The best performing feature (with id=13) corresponds to character n-grams with tf-idf normalization.

classifier	Microsoft		Google	
	score	feature id	score	feature id
Naive Bayes	0.39	12	0.31	12
KNN	0.30	11	0.24	12
SVM (linear)	0.44	13	0.34	13
SVM (RBF)	0.19	14	0.18	14
Random Forest	0.28	3	0.21	3
MLP	0.44	13	0.34	13

in line with other research in this area [9, 10] and adhering to the results of this test, the linear SVM was used for further experiments.

3.4 Experiment Setup

As argued previously, the texts translated by the two translation services have not been mixed for all following calculations, i.e., we never test documents translated by Google on models that have only seen documents translated by Microsoft and vice versa.

To answer both RQs, two separate experiments were conducted: In the first setup, the influence of the number of possible source languages on the resulting accuracy is analyzed. At first, a classification process has been performed on the *short* documents, including all source languages. Subsequently, each combination of 2, 3 and 4 possible source languages has been selected, whereby each combination also contains original, untranslated English documents. The restriction that the combinations must contain the English documents ensures that each combination also contains a non-translated set of documents. This way, any difference in classifying translated vs. original text can be determined as well.

The second experiment subsequently analyzes the influence of the document length on the classification performance. Here, all possible languages are taken into account, resulting in a 10-class classification problem.

4 Results

To quantify the individual expressiveness of each feature, we first conducted isolated evaluations for each feature using the S-dataset. The corresponding results can be found in Table 4. For readability reasons, only selected features are shown in the figure, with the best performing feature printed in bold face. Then, the best performing features were selected, and combinations thereof were tested. Results of these experiments are displayed in Table 5, which displays the best performing feature set (character n-grams with tf-idf normalization, i.e., utilizing combined character 1-, 2-, 3- and 4-grams) including other well-performing combinations. Comparing these results to the previous

single-feature evaluation of Table 4, it can be seen that combining features (sets) does not improve the classification performance.

Table 4. Accuracy of all Single Features

feature id	1	2	3	4	5	6	7	8	9	10	11	12
Microsoft	0.21	0.36	0.33	0.25	0.35	0.39	0.39	0.41	0.29	0.36	0.40	0.41
Google	0.15	0.28	0.25	0.21	0.28	0.29	0.30	0.33	0.23	0.29	0.30	0.31
feature id	13	14	15	16	17	18	19	20	21	22	23	
Microsoft	0.44	0.20	0.27	0.25	0.22	0.27	0.20	0.27	0.25	0.21	0.24	
Google	0.34	0.18	0.24	0.24	0.22	0.26	0.18	0.24	0.23	0.22	0.24	

Table 5. Accuracy of Best Feature and Selected Combinations

features used	accuracy	
	Microsoft	Google
char-n (tfidf)	0.44	0.34
char-n (tfidf) + POS-n (tfidf)	0.25	0.24
char-n (tfidf) + char-n	0.44	0.34
char-n (tfidf) + char-4 (tfidf)	0.42	0.34
char-n (tfidf) + char-4	0.41	0.30

Table 6. Part of the Confusion Matrix for Short Documents Translated by Google

	de	en	es	nl
de	57	24	23	39
en	7	178	4	11
es	28	7	86	17
nl	26	15	26	72

The overall performance on classifying the original language of a machine translated text is high. Fig. 2 shows the classification performance with a varying amount of possible source languages. It can be seen that for all text lengths tested, the prediction is well above the baseline, which represents random guessing out of the possible languages. In case of only two possible source languages, text translated by Microsoft can be classified with an accuracy of over 0.85 when having only short documents as input. The combination of character 1- to 4-grams with tf-idf normalization proves to be the best suiting feature set for both the Microsoft and Google translations.

As expected and displayed in Figure 3, the quality of the predictions also increases substantially with the amount of text available. For the XL-dataset, the accuracy of the classifier reaches values of over 0.99. This result suggests that if the amount of text translated by an engine is large enough, i.e., more than 1,000 words, its unique properties can be used to predict the source language nearly perfectly.

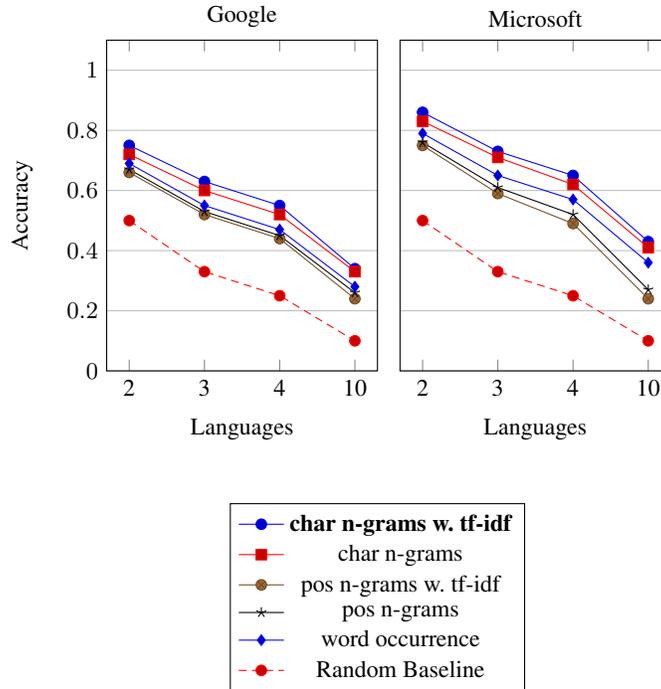


Fig. 2. Varying Number of Possible Languages using short document lengths

Both experiments show that character n-grams with tf-idf normalization achieve the best performance, regardless of the translation engine used. Another fact that can be extracted from the results is that the classification performs worse on text translated by Google rather than by Microsoft. It suggests that Google is currently able to transfer more complex linguistic structures onto its translation than Microsoft does. Note that in general all results are based on the technology currently used by the two translation engines, which may change in the future. While it can be assumed that the prediction accuracy decreases with improvements of the tools, future work should investigate this problem in detail.

Table 6 shows an excerpt of the confusion matrix of short documents using the Google translation service. Due to the origin of the languages it seems intuitive that some pairs (e.g., es-en) are easier distinguishable than others (e.g., nl-de), which is also reinforced by our experiments. However and unfortunately, the SVM used for

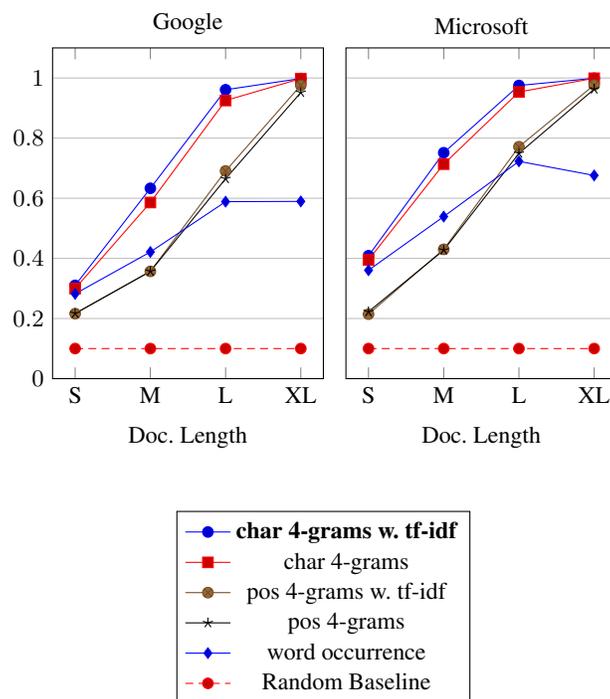


Fig. 3. Varying Document Lengths using 10 languages

the classification provides no information about the relevance of the single character n-grams features, so we can't argue which parts of the translated text indicates the original language at this point. Note that while other classifiers indeed can provide such information (e.g., decision trees or naive Bayes), their prediction accuracy is significantly worse, making statements about the indicators unreliable.

Answering RQ1, the experiments clearly show that it can be determined whether an English text is machine-translated, and that the source language can be predicted with high accuracy. Moreover, the prediction quality increases significantly by restricting the amount of possible source languages as well as by providing longer texts (RQ2).

5 Conclusion and Future Work

In this paper, we have shown that, given an English text, it can be determined if it was translated by using stylometric features in combination with machine-learning. Moreover, if translation was applied, the source language can be predicted at a high accuracy for regular length texts, and at a near-perfect accuracy when providing longer text samples. If the number of possible source languages is low, the prediction is reliable even for short length documents. Possible extensions of this study include considering more different language classes (e.g., Arabic and African), analyzing other target languages for

the translation or finding optimal parameters for different classifiers (e.g., Neural Networks). However, while we showed that source language detection is possible at high accuracy in general, it remains unclear what the discriminating features (i.e., n-grams) are for the respective languages. Using different classifiers which contain feature importances could be used to gain an intuitive understanding of the differences of specific language pairs. Other approaches can include automatic measures like recursive feature elimination, which was omitted due to a limited timeframe. Different kinds of features, such as grammar based features, may help to gain further understanding of the translation systems analyzed, which can subsequently be leveraged to increase the performance of the model ever further.

References

1. Roei Aharoni, Moshe Koppel, and Yoav Goldberg. 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, pages 289–295, Baltimore, USA, June 2014. Association for Computational Linguistics.
2. Serhiy Bykh and Detmar Meurers. Native Language Identification Using Recurring N-grams – Investigating Abstraction and Domain Dependence. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING'2012)*, pages 425–440, Mumbai, India, December 2012.
3. Serhiy Bykh and Detmar Meurers. Exploring syntactic features for native language identification: A variationist perspective on feature encoding and ensemble optimization. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING'2014)*, pages 1962–1973, Dublin, Ireland, August 2014.
4. Serhiy Bykh and Detmar Meurers. Advancing Linguistic Features and Insights by Label-informed Feature Grouping: An Exploration in the Context of Native Language Identification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING'2016)*, pages 739–749, December 2016.
5. Aylin Caliskan and Rachel Greenstadt. Translate Once, Translate Twice, Translate Thrice and Attribute: Identifying Authors and Machine Translation Tools in Translated Text. In *Proceedings of the 6th International Conference on Semantic Computing (ICSC'2012)*, pages 121–125. IEEE, September 2012.
6. Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'2012)*, 2012.
7. Radu Tudor Ionescu and Marius Popescu. Can string kernels pass the test of time in Native Language Identification? In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 224–234. Association for Computational Linguistics, September 2017.
8. Radu Tudor Ionescu, Marius Popescu, and Aoife Cahill. Can characters reveal your native language? A language-independent approach to native language identification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'2014)*, pages 1363–1373. Association for Computational Linguistics, October 2014.
9. Thorsten Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *Proceedings of the 10th European Conference on Machine Learning (ECML'1998)*, pages 137–142. Springer Berlin Heidelberg, April 1998.

10. Moshe Koppel, Jonathan Schler, and Kfir Zigdon. Automatically Determining an Anonymous Author's Native Language. *Intelligence and Security Informatics*, pages 41–76, 2005.
11. Shervin Malmasi and Mark Dras. Language Transfer Hypotheses with Linear SVM Weights. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'2014)*, pages 1385–1390. Association for Computational Linguistics, October 2014.
12. Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Quan. A report on the 2017 native language identification shared task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 62–75. Association for Computational Linguistics, September 2017.
13. Smita Nirkhi and Rajiv V. Dharaskar. Comparative Study of Authorship Identification Techniques for Cyber Forensics Analysis. *International Journal of Advanced Computer Science & Applications*, 4(5), 2013.
14. Daniel Pimienta, Daniel Prado, and Alvaro Blanco. Twelve Years of Measuring Linguistic Diversity in the Internet: Balance and Perspectives. 2009.
15. Martin Potthast, Matthias Hagen, and Benno Stein. Author Obfuscation: Attacking the State of the Art in Authorship Verification. In *Working Notes Papers of the CLEF 2016 Evaluation Labs*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, September 2016.
16. Josyula R. Rao and Pankaj Rohatgi. Can Pseudonymity Really Guarantee Privacy? In *Proceedings of the 9th Conference on USENIX Security Symposium*, volume 9 of *SSYM'00*, pages 7–7. USENIX Association, August 2000.
17. E. Stamatatos. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556, March 2009.
18. Ben Swanson and Eugene Charniak. Extracting the native language signal for second language acquisition. In *Proceedings of NAACL-HLT*, pages 85–94. Association for Computational Linguistics, June 2013.
19. Ben Swanson and Eugene Charniak. Data driven language transfer hypotheses. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: short papers, pages 169–173, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.