Translated Texts Under the Lens: From Machine Translation Detection to Source Language Identification

Massimo La Morgia(✉), Alessandro Mei(✉), Eugenio Nerio Nemmi(✉), Luca Sabatini, and Francesco Sassi(✉)

Sapienza University of Rome, Rome, Italy
{lamorgia,mei,nemmi,sassi}@di.uniroma1.it

Abstract. Machine Translation Systems are today used to break down linguistic barriers. People from different countries and languages can now interact with each other thanks to state-of-the-art translators from prominent software companies like Google and Microsoft. However, these tools are also used to expand the audience for phishing attacks, scam emails or to generate fake reviews to promote a product on different e-commerce platforms. In all these cases, detecting whether a text has been translated can be crucial information. In this work, we tackle the problem of the detection of translated texts from different angles. On top of addressing the classic task of machine translation detection, we investigate and find common patterns across different machine translation systems unrelated to the original text's source language. Then, we show that it is possible to identify the machine translation system used to generate a translated text with high performances (F1-score 88.5%) and that it is also possible to identify the source language of the original text. We perform our tasks over two datasets that we use to evaluate our models: Books, a new dataset we built from scratch based on excerpts of novels, and the well-known Europarl dataset, based on proceedings of the European Parliament.

Keywords: Machine Translation Systems · Machine Learning · Natural Language Processing

1 Introduction

Today, hundreds of thousands of people use commercial machine translation systems (MTSs) worldwide for personal or working purposes. They help bridge the gap in language barriers, especially on the Web, by facilitating communication between people. However, bad actors use these systems to target potential victims of email-phishing [32] massively or generate fake reviews of products to trick recommendation systems [16] and people into buying or choosing a specific product. For all these reasons, machine translation detectors are actively used to infer spam emails or to detect poor quality web pages [13].

In this work, we put automatically translated texts under the lens. We study the impact of the MTSs and the source language of the translated text on the

B. Crémilleux et al. (Eds.): IDA 2023, LNCS 13876, pp. 222–235, 2023.
https://doi.org/10.1007/978-3-031-30047-9_18

Machine Translation Detection (MTD) task leveraging Books, a novel dataset built from excerpts of novels. We find that MTSs have common patterns that can be learned by training on a single MTS; thus, we are able to identify translated text regardless of the MTS used for the translation, suggesting that the automatic translation process introduces recognizable patterns in the translation. Similarly, we discover that we can learn these patterns regardless of the source language of the translated text. We can train on a single MTS using text from a single source language and still detect the translated text on multiple MTSs and source languages with comparable performances.

We then investigate the possibility of identifying the MTS used to produce the translation and the source language of the original text. To explore these questions, we introduce, to the best of our knowledge, two new tasks: Machine Translation Identification (MTI) and Source Language Identification (SLI). In the former (MTI), we want to identify which MTS has been used to generate a translation, while in the latter (SLI), we want to identify the source language of a translated text. For the first task, MTI, we built a classifier that shows promising results, with an average F1-score of 88.5%. In the second task, SLI, we propose a stacked classifier able to identify the source language of a machine-translated text with an average F1-score of 78% among 4 languages. We believe that these tasks could be helpful in forensic analysis, where malicious actors attempt to obfuscate their writing style using MTSs [17,25]. In particular, in this paper, we try to answer the following research questions:

Q1. Is it possible to identify a translated text regardless of the MTS used or the source language of the text?
Q2. Is it possible to identify which translator has been used to translate the text?
Q3. Is it possible to recognize the source language of the translated text?

2 Datasets

Since we need specific information to explore our questions, we build new datasets. Indeed for Q1 and Q2, we need the translation of an *original* sample both by a human and an MTS, while for Q3, we need to know the source language. In particular, to assess our experiments over different settings and topic domains, we perform our study using two datasets: one extracted from novels and the other based on speech transcriptions. The first dataset we use is *Books*, a novel dataset we introduce. To build Books, we collect 100 books originally written in 4 different languages by 100 different established writers of the XX/XXI century [37]. In particular, we select 25 books for each of the following source languages: Italian, French, Spanish, and German. The selected books belong to several different domains and authors. Thus, they have very different writing styles. From each book, we select an excerpt of approximately 10,000 characters (on average 1642.67 words per novel) and their corresponding translation from the English edition. Finally, we produce 3 more English translations for each original excerpt using the APIs of 3 state-of-the-art commercial

Machine Translation Systems: Google Translate [12] (GT), Microsoft Translation [26] (MT), and DeepL [7] (DL). At the end of the process, the Books dataset is made of 400 different samples.

The second dataset we use for our experiments is Europarl [18]. It is a parallel corpus extracted from the proceedings of the European Parliament containing *speech transcripts* of European parliamentarians and the corresponding professional translations into each of the 20 European languages. The texts on this dataset include many speech-distinctive elements such as hesitations, broken sentences, and repetition [5]. Consistently with Books, we obtain 100 seed samples by extracting from Europarl 25 samples for each of the 4 languages we consider. Every sample is made using transcripts of speakers of the same source language and contains about 10,000 characters (on average 1512.81 words per sample). We pre-process the dataset using Moses [19], a statistical machine translation system that includes different tools and utilities to parse and parallelize the Europarl dataset. Then, we collect the parallel English translation of each seed sample. Finally, we translate each seed sample using the selected MTSs. Figure 1 summarizes the process of building the Books and Europarl dataset. Both datasets at the end contain 400 samples in English, which are produced starting from 100 seeds (25 for each language), of which 100 were made by translating the original seed by professional human translators and 300 using machine-translation systems (100 for each MTS).

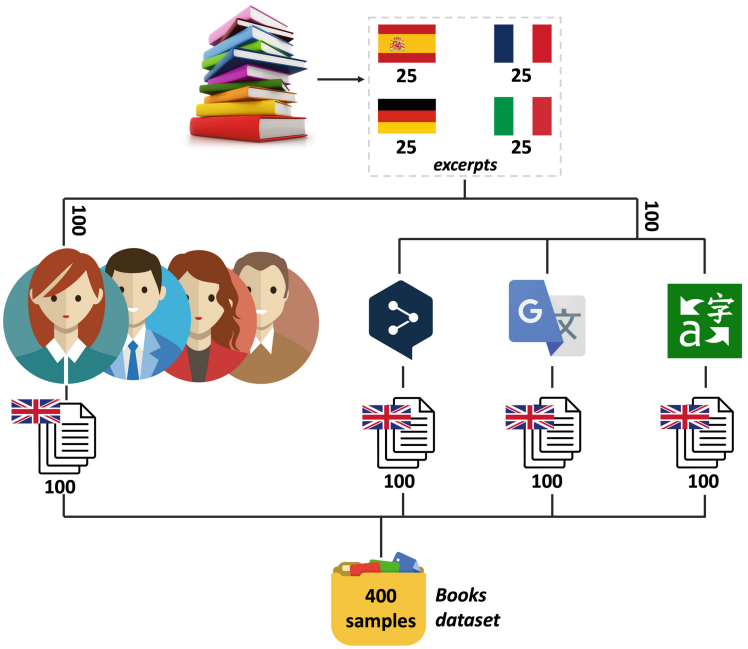

Fig. 1. Step by step representation of the process used to build Books dataset. The same pipeline was applied to build the Europarl dataset.

3 Experimental Settings

In this section, we describe the experimental settings of the tasks in terms of train/test splits, the pre-processing we apply, and the features we use. For all the experiments, we use 60% of the dataset as train and 40% as test. We use Python Scikit-learn [33] to implement all the models and the feature selection techniques. Whenever the model parameters are not specified, we use the default values.

3.1 Pre-processing and Feature Description

We apply three pre-processing techniques to extract our features. Firstly, we tokenize the texts. Tokenization is the process of separating a piece of text into smaller units called tokens (*e.g.,* words, char). We then apply the following processes:

- ***Stemming.*** It is the process of reducing inflected words to their root (words stem). (*e.g.,* writing → write; eating → eat)
- ***Part-of-Speech (POS) tagging.*** It is the process of identifying a word's appropriate part of speech in a text based on its definition and context.
- ***Distortion Text.*** It is a process where ASCII characters are replaced with a special character [36]. Table 1 shows an example of this pre-processing step.

Table 1. Example of text distortion.

Original Text	Distorted Text
I don't know. Just making conversation with you, Morty. What do you think, I-I-I... know everything about everything?	* ***'* ****. **** ****** ********** **** ***, *****. **** ** *** *****, *-*-*... **** ********** ***** **********?

Most of the features are based on *n-grams*, that are a sequence of N contiguous elements, in our case, character (char-gram) or words (word-gram). We use the notation *Char-gram (i-k)* (resp. *Word-gram (i-k)*) to denote all the char n-grams (resp. word n-grams) with $n \in \{i, \ldots, k\}$. Table 2 shows the features we use for our tasks and the feature number for the different tasks and datasets. Below a description of each feature:

- ***Char-gram*** is a sequence of N contiguous characters.
- ***Sentence Length*** is the average length of the sentences for each text based on the number of characters.
- ***Words avg*** is the average number of words for each sentence of the text.
- ***Adjectives avg*** is the average number of adjectives for each text.
- ***Dist Char-gram (i-k)*** are char-grams computed over the distortion text.
- ***POS Word-gram(i-k)*** are word-grams computed over Part of Speech (POS) tagged text.

– ***Type Token Ratio*** (*TTR*) is the ratio between the number of unique words and the total number of words for a given text. The idea behind this feature is to measure the vocabulary variety (in terms of words) of a text.

We use the Bag of Words to weigh the char-grams and word-grams, while we use Tf-idf (term frequency-inverse document frequency) to weigh the distortion text. The Bags of Words is a representation that creates vectors with the number of occurrences of a specified element in the text (*e.g.,* words), while the Tf-idf is an weighting schema that gives a larger value (weight) to elements that are less frequent in the document corpus.

Table 2. Features types and numbers of features for the MTI and SLI tasks on both datasets.

Feature Type	MTI		SLI	
	Books	Euro	Books	Euro
Char-gram (1–6)	318,250	220,593	261,895	175,247
Sentence Length	1	1	1	1
Words avg	1	1	1	1
Adjectives avg	1	1	1	1
Dist. Char-gram (5–8)	15,134	12,080	–	–
Dist. Char-gram (2–8)	–	–	13,897	9,522
POS Word-gram (1–6)	–	–	187,481	145,473
TTR	1	1	–	–
All	333,388	232,677	463,276	330,245

4 Machine Translation Detection

The goal of the *Machine Translation Detection* (*MTD*) task is to automatically detect whether a text has been translated by a machine translation system or is human-generated. This task was broadly studied in the literature with different approaches such as using fixed features [1,23], n-gram [2,34], coherence score [27] and similarity with round-trip translation [28]. In this section, we first want to replicate similar results to the state-of-the-art on our datasets Books, to verify that it is suitable for our purposes. Then, we design two experiments to explore further the underlying patterns of machine-translated texts.

For all the experiments in this section, we use the following model. We train a Multilayer Perceptron [15] with a single hidden layer made of 10 neurons and a BFGS optimizer [3] for weights optimization. Regarding the features, we compute all the char n-grams with $n \in \{1, \ldots, 6\}$ and then select the 2,500 more relevant n-grams according to the chi-square metric [9]. We finally normalize the features with the SkLearn StandardScaler. Figure 2 shows the results on Books and Europarl datasets.

We obtain a high F1-score on both corpora (0.9 on Books and 0.97 on Europarl), showing that our model can achieve excellent results in distinguishing machine-translated and human-translated texts.

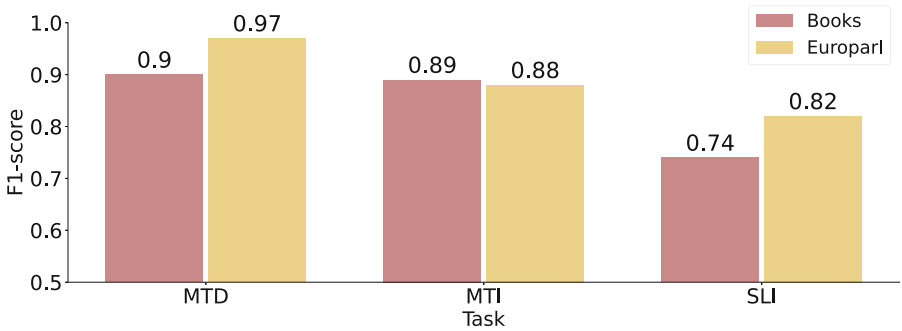

Fig. 2. F1-score for the Machine Translation Detection (MTD), Machine Translation Identification (MTI), and Source Language Identification (SLI) tasks on the Books and Europarl datasets.

4.1 Learning from a Single MTS

The next interesting point to explore is if there are any common patterns among the different MTSs that could be learned to identify a translated text, even if it is generated by an MTS that was not included in the training set. To verify this idea, we train our model using only samples translated by a single MTS and human-translated samples. We repeat the experiment 3 times, training the model at each iteration with samples produced by a different MTS and testing it only on the samples of the remaining MTSs.

Table 3(a) shows the results of this experiment for the different combinations. As we can see, the model is able to achieve good results (on average 88% of F1-score) when tested on samples generated by machine translators that are not represented in the training set. Interestingly, the model trained on MT achieves similar (average delta 0.015) results to those obtained by training the model using the whole dataset (*i.e.,* training on all the MTSs).

These results suggest that there are some common patterns among the MTSs that the model can learn from a single MTS.

4.2 Learning from a Single Language

Since we have 4 different source languages in our dataset, we want to understand the impact they might have on the MTD task. In this experiment, we train our model using only the translation from one source language and test it against the sample produced by the translation from other source languages and the human-translated samples. Table 3(b) shows the F1-score using the different source languages.

Table 3. 3(a): F1-score for Task 1 training on a single MTS' samples and testing on the others. 3(b): F1-score for Task 1 training on a single language and testing on the others.

3(a) Task 1 - single MTS

Train	Books	Europarl
GT	0.85	0.82
MT	0.89	0.95
DL	0.84	0.94

3(b) Task 1 - single language

Train	Books	Europarl
IT	0.91	0.93
FR	0.85	0.74
ES	0.88	0.78
DE	0.73	0.81

Results show that the model can learn machine translation patterns even when training only on one language, suggesting that these patterns are unrelated to the source language but rather unique to the machine translation process.

5 Machine Translator Identification

Results from the previous section suggest common patterns exist among the different MTSs that allow us to differentiate machine-translated texts from human-translated ones. In this section, we investigate if MTSs translations differ enough from each other to be able to identify which one has been used to translate a sample (Question Q2). Thus, given a machine-translated text T', our goal is to identify the MTS M that generated the text T'. We call this task ***Machine Translator Identification*** (*MTI*). In particular, we focus on identifying the 3 MTSs used to build the Books and Europarl datasets: *Google Translate*, *Microsoft Translation*, and *DeepL*. Given the task's goal, we use a sub-set of Europarl and Books datasets for the following experiments, removing the 100 samples representing the class of human translations from each dataset.

For this task, we build an ensemble classifier. The first level comprises three different classifiers: a Support Vector Machine, a Logistic Regression, and a Random Tree. Then, the outputs of the classifiers are used as input to feed a hard voting layer (SkLearn VotingClassifier) for the final prediction. Table 2 shows the type and the number of features we use to train the three classifiers at the first level of our architecture. For all the n-gram type features, we select only the 85% most significant ones using SelectPercentile of SkLearn, and we standardize them with the SkLearn StandardScaler. Figure 2 reports the F1-score for the two datasets. As we can notice, our classifier performs similarly on both datasets, with an F1-score of 0.89 and 0.88 for Books and Europarl, respectively. To better understand the results, we analyze the confusion matrices of the two classifications. The confusion matrix of Books (Table 4) shows that GT is the hardest MTS to identify, and its misclassified samples are mostly assigned to the MT class.

We found a possible explanation for these errors by analyzing the BLUE score [31]. The BLEU Score is an algorithm for evaluating the quality of a trans-

Table 4. Confusion Matrix on Books for the MTI task.

Label \ Predicted	GT	MT	DL
GT	30	6	4
MT	1	39	0
DL	0	2	38

lation. It measures the similarity of the translation to a reference one. For each pair of the MTSs, we measure the BLEU score, obtaining a value of 69 for the pair GT-MT, 63 for GT-DL, and 62.4 for DL-MT (Table 5).

Table 5. BLEU score for the MTS pairs.

	MT	DL	GT
GT	69	63	–
DL	62.4	–	–

The high BLEU score between GT and MT shows that they have similar translations, which could be the reason for the incorrect classification of the GT samples. Conversely, the low similarity between the MT and DL classes could lead to the high accuracy we observe in our experiment. Finally, we obtain similar results by analyzing the confusion matrix and the BLUE score for the Europarl dataset.

6 Source Language Identification

As a final task, we propose the ***Source Language Identification*** (SLI). The goal of the task is to identify the source language of a given machine-translated text. Thus, given a machine-translated text T' in a language $L2$ (in our case English), the goal of the task is to identify the language $L1$ of the text T. This task could be considered a variation of other tasks already studied in the literature, such as Native Language Identification (NLI), where the goal is to identify the native language (L1) of a person who writes in another language (L2) or determining the source language of a human-translated text (see Sect. 7), where the goal is to identify the source language of a text that has been human-translated. However, unlike the previous studies, our task focuses on identifying the source language of a text that is translated by a Machine Translation System and not by a human. For our experiments, we consider English as $L2$, and the possible $L1$ languages are: Italian, French, Spanish or German. Since we only care about translations of MTS (*i.e.,* text not translated by human), we modify our dataset in the same way as we did for the MTI task (Sect.5).

For this task, we use the stacking ensemble technique. In particular, we stacked an AdaBoost [10] model with 50 LinearSVC [6] and a Logistic Regression [39] model as base estimators. Table 2 shows the type and the number of features we use to train the stacking classifier. For all the n-gram features, we select the top 70% according to their F-value, computed with the variance analysis (ANOVA) [35]. Then, we standardize them with a StandardScaler. Figure 2 shows the F1-score of the model trained and tested on both our datasets. The results suggest that identifying the source language is easier in Europarl than in Books. As noted in [14], a possible reason could be that the Europarl dataset may contain some distinctive patterns for the source language of the speaker since it is a transcription of a talk. Instead, the Books dataset covers a wide area of topics and contains fewer clues about the author's source language. Table 6 shows the confusion matrices on the Books and Europarl dataset.

Table 6. Confusion Matrices on Books and Europarl for the SLI task.

Books

Label \ Predicted	DE	ES	FR	IT
DE	25	0	5	0
ES	2	23	4	1
FR	0	2	27	1
IT	1	5	9	15

Europarl

Label \ Predicted	DE	ES	FR	IT
DE	27	3	0	0
ES	0	24	3	3
FR	0	3	24	3
IT	0	9	1	20

The most challenging source language to detect on both datasets is Italian, frequently misclassified as Spanish or French. German is generally better identified than the other languages except for French on the Books dataset, with five classification errors. Indeed, German has the highest F1-score among all the classes, with a value of 0.86 in Books and 0.94 in Europarl. This is intuitive and expected, since German is a West Germanic language while the other 3 are Romance languages and have more features in common [30].

7 Related Work

Machine Translation Detection: The detection of automatic translations has been investigated in the past using multiple techniques. Both Aharoni et al. [1] and Li et al. [23] use fixed features taken from the English language that may be used regardless of the language in which the content was originally written (*i.e.,* source language). They respectively achieve an accuracy of 90% and 83%. Arase et al. [2] and Popescu et al. [34] use an n-gram based approach to perform the task, reaching a high accuracy of 96% and 99%. Other works used words distribution [22,29], that lead to a max accuracy of 98%, or coherence score [27], with an accuracy of 73%. More recently, Nguyen et al. [28] propose a method

to detect translated texts using text similarity with round-trip translation that appears to be resistant for different translators and languages. In this case, they were able to achieve an accuracy of 94%.

Machine Translator Identification: Bhardwaj et al. [4] test 18 classifiers to detect translated text using commercial as well as in-house MTS. Looking at the identification of the MTS, previous works [1] show that testing machine translation detection over different MTSs produces different results. This suggests that these MTSs have different qualities of translation and that there are differences between them. In the same way, Bizzoni et al. [5] found similar results studying translationese ([11]) over different architectures. These studies show that there could be enough differences in MTS systems to be able to identify which translator has been used for a given translation.

Source Language Identification: We can have three slightly different settings for the source language identification task. The first setting is the well-known NLI task [38] where the goal is to identify the native language *L1* of a person who writes a text in a second language *L2* [21]. The second setting is when the translation has been performed by a person that is different from the one that wrote the original text. In [14], the author shows that it is possible to identify the source language of the translation of speeches in the Europarl corpus with an accuracy of more than 87%, without testing if these results hold for translated text (i.e. it is possible to detect the source language of an automatically translated text). Using human translation, also Lynch et al. [24] and Koppel et al. [20] perform the same task showing that it is possible to determine the original language of a human translation.

8 Conclusion and Future Work

In this work, we put translated text under the lens. We start by evaluating the impact of MTSs and source languages on the Machine Translation Detection task. We find that MTSs generate common patterns in the translated text that can be learned by a machine learning model trained using a single MTS. Furthermore, we show that the performance of the task is not significantly influenced by the MTS employed or the source language of the text. These results open the possibility to employ machine learning models trained solely on a subset of known MTSs or languages and identify text translated from any other MTSs or languages. Then, to the best of our knowledge, we introduce two new tasks: Machine Translator Identification and Source Language Identification. The goal of the Machine Translator Identification task (MTI) is to identify the MTS that has been used to translate a target text, while the Source Language Identification (SLI) task aims to identify the source language of a machine-translated text.

The models we propose for both tasks achieve an average F1-score of 88.5% and 78%, respectively, for the MTI and the SLI task. These last two tasks can help to characterize translated texts further and could be used as features for

a classification task or give additional insights when studying potential threats. Our results, although they represent a first attempt to tackle the newly presented tasks, show that much more work can be done in this area.

While we achieve good performances, we believe there could be further improvement by using deep learning models that are particularly effective in NLP tasks, such as BERT, a pre-trained language representation model based on transformers [8]. However, the number of samples in the datasets should be increased to use deep learning techniques effectively. Furthermore, in our study, we perform all the experiments at the document level, using a mean of 1642.67 words. In the future, it would be interesting to propose the same tasks in a more challenging setting, using sentences rather than documents. This is particularly important since it makes it possible to evaluate very short texts. We consider only European languages (although with different origins) for the datasets: German, French, Italian, and Spanish. However, there are other languages, such as Arabic, Mandarin, or Hindi, that are widely used worldwide, and it could be interesting to expand the datasets and test the classifiers performances with the new data. Finally, with the recent popularity of Large Language Models (LLMs) such as ChatGPT, it could be interesting to verify if our model can still identify a text translated by ChatGPT and its original language, and also to introduce a new task for the detection of text generated by LLMs.

Acknowledgements. This work was supported in part by the MIUR under grant "Dipartimenti di eccellenza 2018-2022" of the Department of Computer Science of Sapienza University.

References

1. Aharoni, R., Koppel, M., Goldberg, Y.: Automatic detection of machine translated text and translation quality estimation. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 2: Short Papers), pp. 289–295. Association for Computational Linguistics, Baltimore (2014). https://doi.org/10.3115/v1/P14-2048, https://aclanthology.org/P14-2048
2. Arase, Y., Zhou, M.: Machine translation detection from monolingual web-text. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers), pp. 1597–1607. Association for Computational Linguistics, Sofia (2013). https://aclanthology.org/P13-1157
3. Battiti, R., Masulli, F.: Bfgs optimization for faster and automated supervised learning. In: International Neural Network Conference, pp. 757–760. Springer, Dordrecht (1990). https://doi.org/10.1007/978-94-009-0643-3_68
4. Bhardwaj, S., Alfonso Hermelo, D., Langlais, P., Bernier-Colborne, G., Goutte, C., Simard, M.: Human or neural translation? In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 6553–6564. International Committee on Computational Linguistics, Barcelona (2020). https://doi.org/10.18653/v1/2020.coling-main.576, https://aclanthology.org/2020.coling-main.576
5. Bizzoni, Y., Juzek, T.S., España-Bonet, C., Dutta Chowdhury, K., van Genabith, J., Teich, E.: How human is machine translationese? comparing human and machine translations of text and speech. In: Proceedings of the 17th International

Conference on Spoken Language Translation, pp. 280–290. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.iwslt-1.34, https://aclanthology.org/2020.iwslt-1.34
6. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
7. DeepL: Deepl translator (2021). https://www.deepl.com/pro-api
8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
9. Forman, G., et al.: An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. **3**, 1289–1305 (2003)
10. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997)
11. Gellerstam, M.: Translationese in swedish novels translated from English. In: Wollin, L., Lindquist, H. (eds.) Translation Studies in Scandinavia: Poceedings from the Scandinavian Symposium on Translation Theory (SSOTT) II, pp. 88–95. no. 75 in Lund Studies in English, CWK Gleerup, Lund (1986)
12. Google: Google translator (2021). https://cloud.google.com/translate
13. Google: Managing multi-regional and multilingual sites (2021). https://developers.google.com/search/docs/advanced/crawling/managing-multi-regional-sites
14. van Halteren, H.: Source language markers in EUROPARL translations. In: Proceedings of the 22nd International Conference on Computational Linguistics (2008), pp. 937–944. Coling 2008 Organizing Committee, Manchester, UK (2008). https://aclanthology.org/C08-1118
15. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Netw. **2**(5), 359–366 (1989)
16. Juuti, M., Sun, B., Mori, T., Asokan, N.: Stay on-topic: generating context-specific fake restaurant reviews. In: Lopez, J., Zhou, J., Soriano, M. (eds.) ESORICS 2018. LNCS, vol. 11098, pp. 132–151. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99073-6_7
17. Kacmarcik, G., Gamon, M.: Obfuscating document stylometry to preserve author anonymity. In: Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pp. 444–451. Association for Computational Linguistics, Sydney (2006). https://aclanthology.org/P06-2058
18. Koehn, P.: Europarl: a parallel corpus for statistical machine translation. In: Proceedings of Machine Translation Summit X: Papers, pp. 79–86. Phuket, Thailand (2005). https://aclanthology.org/2005.mtsummit-papers.11
19. Koehn, P., et al.: Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, pp. 177–180. Association for Computational Linguistics, Prague (2007). https://aclanthology.org/P07-2045
20. Koppel, M., Ordan, N.: Translationese and its dialects. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 1318–1326. Association for Computational Linguistics, Portland (2011). https://aclanthology.org/P11-1132
21. La Morgia, M., Mei, A., Nemmi, E., Raponi, S., Stefa, J.: Nationality and geolocation-based profiling in the dark (web). IEEE Trans. Serv. Comput. **15**(1), 429–441 (2019)

22. Labbé, C., Labbé, D.: Duplicate and fake publications in the scientific literature: how many SCIgen papers in computer science? Scientometrics **94**(1), 379–396 (2013). https://doi.org/10.1007/s11192-012-0781-y
23. Li, Y., Wang, R., Zhao, H.: A machine learning method to distinguish machine translation from human translation. In: Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation: Posters, pp. 354–360, Shanghai, China (2015). https://aclanthology.org/Y15-2041
24. Lynch, G., Vogel, C.: Towards the automatic detection of the source language of a literary translation. In: Proceedings of COLING 2012: Posters, pp. 775–784. The COLING 2012 Organizing Committee, Mumbai, India (2012). https://aclanthology.org/C12-2076
25. Mahmood, A., Ahmad, F., Shafiq, Z., Srinivasan, P., Zaffar, F.: A girl has no name: Automated authorship obfuscation using mutant-x. Proc. Priv. Enhancing Technol. **2019**(4), 54–71 (2019)
26. Microsoft: Microsoft translator (2021). https://www.microsoft.com/translator/
27. Nguyen-Son, H.Q., Nguyen, H.H., Tieu, N.D.T., Yamagishi, J., Echizen, I.: Identifying computer-translated paragraphs using coherence features. In: Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation. Association for Computational Linguistics, Hong Kong (2018). https://aclanthology.org/Y18-1056
28. Nguyen-Son, H.Q., Thao, T., Hidano, S., Gupta, I., Kiyomoto, S.: Machine translated text detection through text similarity with round-trip translation. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5792–5797. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.naacl-main.462, https://aclanthology.org/2021.naacl-main.462
29. Nguyen-Son, H.Q., Tieu, N.D.T., Nguyen, H.H., Yamagishi, J., Zen, I.E.: Identifying computer-generated text using statistical analysis. In: 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1504–1511. IEEE (2017)
30. Padró, M., Padró, L.: Comparing methods for language identification. Procesamiento del lenguaje natural **33** (2004)
31. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics, Philadelphia (2002). https://doi.org/10.3115/1073083.1073135, https://aclanthology.org/P02-1040/
32. Parmar, Y.S., Jahankhani, H.: Utilising machine learning against email phishing to detect malicious emails. In: Montasari, R., Jahankhani, H. (eds.) Artificial Intelligence in Cyber Security: Impact and Implications. ASTSA, pp. 73–102. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88040-8_3
33. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
34. Popescu, M.: Studying translationese at the character level. In: Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, pp. 634–639. Association for Computational Linguistics, Hissar (2011), https://aclanthology.org/R11-1091
35. St, L., Wold, S., et al.: Analysis of variance (ANOVA). Chemom. Intell. Lab. Syst. **6**(4), 259–272 (1989)

36. Stamatatos, E.: Authorship attribution using text distortion. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Vol. 1, Long Papers, pp. 1138–1149 (2017)
37. SystemsLab: Book dataset. https://github.com/SystemsLab-Sapienza/books-dataset
38. Tetreault, J., Blanchard, D., Cahill, A.: A report on the first native language identification shared task. In: Proceedings of the 8th Workshop on Innovative use of NLP for Building Educational Applications, pp. 48–57 (2013)
39. Wright, R.E.: Logistic regression (1995)