Integrating Back-Translation into BERT Model for Detecting Machine-Translated Text

Ishita Gupta[1,a) Hoang-Quoc Nguyen-Son[2,b)] Tran Phuong Thao[3,c)]
Seira Hidano[2,d)] Shinsaku Kiyomoto[2,e)]

Abstract:
Machine-generated text is being used by adversaries to support malicious purposes like spam mails and fake reviews etc. Recently a form of back translation plagiarism has started where texts are paraphrased by translating into a different language and then back into the original language. Previous methods for detecting such machine-generated text are based only on the intrinsic content of the text. We propose a detector which exploits the external information obtained from back-translation; and integrates it into the BERT model. An evaluation of 90000 samples of original English sentences and translated French sentences shows that our detector can classify them with 83.8% accuracy. This is higher than previous methods whose best accuracy is 79.9%. Moreover, our detector can efficiently detect back-translated text with 87.1% accuracy when assessed on 20000 sentences. This is an improvement from 82.2% accuracy of the state of the art. We have also conducted experiments with low-resource language and reached similar results. This demonstrates the persistence of our detector on various tasks in both rich- and low-resource languages.

Keywords: Machine-Translation Detection, Back-Translation, BERT Model, Paraphrasing, Adversarial Text

1. Introduction

Nowadays, cross-language communication among people plays an important role in modern life. It opens great opportunities in various fields such as entertainment, e-commerce, career, etc. In this communication, a machine translator is an essential component. Moreover, the translator can also support other mutual interactions among machines and between a human with a machine. For example, a new AI system can be built from the other mature systems, which are operated in another language. In another example, the cutting-edge smart devices such as Apple Siri and Google Home have already supported multiple languages via translators.

However, the main problem of using translation is that it can lead to misunderstanding due to the diversity of language usages such as slang, idiom, dialect, etc. In another problem, adversaries can take advantage of translators to generate paraphrasing texts for malicious purposes, for example, plagiarism [1] and style transfer [2]. Spreading such artificial texts can seriously reduce the reputation of the original texts which are created from human society. Therefore, it is crucial to develop a detector for determining whether a text is written by a human or generated by a translator.

Many researchers have interested in detecting machine-translated text. The most common methods are based on the N-gram model [3], [4], [5] to measure the fluency of text. On the other hand, the structure of the parsing tree is exploited to recognize the machine-generated texts [6], [7]. Moreover, the different word usages in human and machine texts lead to the differences in their word distributions [8]. Other researchers prove that the coherence of the human-written text is better than the machine-translated one [9], [10]. Beyond detecting machine-translated text, artificial fake reviews and papers are also recognized by readability [11] and duplicate patterns [12], respectively. Methods to detect such machine translation plagiarism are often based on identifying certain vocabulary token frequencies [13]. More recently, deep learning networks have been used to improve the accuracy of many NLP tasks like paraphrase detection and sentiment identification [14]. Deep learning can be used to learn context and word patterns making it ideal for detecting machine-translated text by learning the difference in the sentence structure of the computer-generated and human-generated text.

The limitation in all existing methods above is that they only analyze the intrinsic contents of machine-generated texts but ignore the original processes which are used to produce the texts.

Our idea based on the fact that the processing on original

[1] Indian Institute of Technology, Delhi, India
[2] KDDI Research Inc., Saitama, Japan
[3] The University of Tokyo, Tokyo, Japan
[a)] Ishita.Gupta.ee317@ee.iitd.ac.in
[b)] ho-nguyen@kddi-research.jp
[c)] tpthao@yamagula.ic.i.u-tokyo.ac.jp
[d)] se-hidano@kddi-research.jp
[e)] kiyomoto@kddi-research.jp

***h*$_0$**: *"When we have finalised our proposal on the new rules and decided on the most suitable legal form, I will be happy to present our viewpoint to you."*
***h*$_1$**: *"Lorsque nous aurons finalisé notre proposition sur les nouvelles règles et décidé de la forme juridique la plus appropriée, il me fera plaisir de vous présenter notre point de vue."*
***h*$_2$**: *"**Once** we have **finalized** our proposal on the new rules and decided on the most **appropriate** legal form, I will be **pleased** to present our **point of view.**"*
***h*$_3$**: *"Une fois que nous aurons finalisé notre proposition sur les nouvelles règles et décidé de la forme juridique la plus appropriée, je me ferai un plaisir de présenter notre point de vue."*
***h*$_4$**: *"Once we have finalized our proposal on the new rules and decided on the most appropriate legal form, I will be **happy** to present our point of view."*
***h*$_5$**: *"Une fois que nous aurons finalisé notre proposition sur les nouvelles règles et décidé de la forme juridique la plus appropriée, je me ferai un plaisir de présenter notre point de vue."*
***h*$_6$**: *"Once we have finalized our proposal on the new rules and decided on the most appropriate legal form, I will be happy to present our point of view."*

Fig. 1 The variants of repeatedly using back-translations.

data often produces more variations than that on modified data. For example, in the field of image, equalizing histogram on an original image makes the much larger change than that on a balanced image, which has already equalized before. In the field of text, we also have a similar phenomenon. More particularly, we conduct a random example of an original sentence h_0 from European parallel corpus[*1] as shown in Figure 1. h_0 is translated to French h_1 and then re-translated to English to create the back-translation denoted as h_2 where the subscript 2 represents to the number times of transitions applied from h_0. The other back-translations h_4 and h_6 are generated in the same manner of h_2. The variation between a back-translation with its origin is highlighted in bold with word usage and in underline with structure. The back-translation reaches the saturation in h_6 with no change. Among back-translations, h_2 has the largest variations with seven positions in the word usage and three positions in the structure. The variations are remarkably reduced in the next back-translation h_4 with only one position in the word usage and nothing in h_6. The example demonstrates that the earlier generations have a higher number of variations than the latter ones.

We check our findings on machine-translated text detection. More particularly, we picked up the English-French pair $\{h_0, m_0\}$ in the parallel corpus in which h_0 is analyzed above. While h_0 is considered as the human-written sentence, m_0 is translated to English by Google for generating a machine sentence m_1 as shown in Figure 2. We then generate the two back-translation versions h_2 and m_3 using French as the intermediate language. While h_2 is translated in two times from the origin h_0, m_3 is generated after three times from m_1. At the result, h_2 has more variations with h_0 in word usage than m_3 with m_1. Moreover, the structure in h_2 is slightly changed whereas in m_3 is preserved. It demonstrates that the differences in back-translation can be used to distinguish human-written with machine-translated text.

In this paper, we have proposed a method using back-translation to detect machine-translated text. Our contributions are listed as below:

- We explore the variant of a text when repeatedly back-translated in the same translator. In particular, the text is invariant after certain times of back-translating. Moreover, the earlier back-translations produce the larger variants than the later ones.
- We suggest a scheme for detection of machine-translated text by integrating back translation features with the corresponding input text. Followed by classification with the use of Bidirectional Encoder Representations Transformer (BERT) network.

We randomly selected 45000 English-French sentence pairs from the European corpus for evaluation. While the English was considered as the human-written text, the French was translated to English using Google and is represented for the machine-translated text. Our method achieves accuracy of 83.8%. It outperforms previous methods with the best accuracy as 79.9%. The similar experiment was conducted with back-translation detection. More specifically, we chose all 11748 sentiment sentences in both negatives and positives from the Stanford Treebank corpus[*2]. We then generated the machine back-translated text using French as the intermediate language. Our method outperforms with 87.1% of accuracy. We conducted further experiments with Japanese and reach similar results. It demonstrates the persistence of the proposed method in various tasks in both low and rich resource languages.

The rest of the paper is organized as follow. Section 2 describes some main previous methods of detecting machine-translated and other machine-generated texts. The proposed method is presented in Section 3. The experimental results are shown in Section 4. Finally, we summarize some main key points and mention future work in Section 5.

2. Related Work

2.1 Machine Translation Detection

The previous methods for detecting machine-translated

[*1] https://www.statmt.org/europarl/

[*2] http://nlp.stanford.edu/~socherr/stanfordSentimentTreebank.zip

	m_0: "*Quand nous aurons mis au point notre proposition sur les nouvelles règles et choisi la forme juridique la plus adaptée, je me ferai un plaisir de vous exposer nos vues.*"
h_0 **(human-written text):** "***When*** *we have* ***finalised*** *our proposal on the new rules and decided on the most* ***suitable*** *legal form, I will be* ***happy*** *to present our* ***viewpoint to you***."	***m_1*** **(machine-translated text):** "***When*** *we have finalized our proposal on the new rules and chosen the most appropriate legal form, I will be happy to share our views.*"
h_1: "*Lorsque nous aurons finalisé notre proposition sur les nouvelles règles et décidé de la forme juridique la plus appropriée, je me ferai un plaisir de vous présenter notre point de vue.*"	***m_2***: "*Lorsque nous aurons finalisé notre proposition sur les nouvelles règles et choisi la forme juridique la plus appropriée, je serai heureux de partager nos points de vue.*"
h_2: "***Once*** *we have* ***finalized*** *our proposal on the new rules and decided on the most* ***appropriate*** *legal form, I will be* ***pleased*** *to present our* ***point of view***."	***m_3***: "***Once*** *we have finalized our proposal on the new rules and chosen the most appropriate legal form, I will be happy to share our views.*"

Fig. 2 Human-written vs machine-translated text.

text can be split into four groups.

2.1.1 *N*-gram model

This model is commonly used to estimate the fluency of continuous words. Researchers have suggested additional features to support the original model. For example, Arase and Zhou [4] estimated the fluency of non-continuous words by sequential pattern mining. They can extract fluent human patterns (e.g., "*not only * but also*," and "*more * than*") comparing with weird machine patterns (e.g., "*after * after the*," "*and also * and*"). On the other hand, Aharoni et al. [3] combined the POS N-gram model with functional words, which abundantly occur in the machine-translated text. Nguyen-Son and Echizen [5] also integrated the word N-gram model with noise features for detecting translation in online social networking (OSN) messages. Such specific features often occur in human messages such as misspelling and spoken words or in machine messages, for example, untranslated words. However, these noises frequently appear in the OSN messages more than others.

2.1.2 Parsing tree

Li et al. [6] used the syntactic parsing tree for classifying human and machine sentences. They claim that the structure of a human parsing is more balancing than that of a machine. They thus extracted balancing-based features such as the ratio between left and right nodes in both general and main continents. The limitation of this approach is that it ignores the semantic meaning of the text.

2.1.3 Word distribution

The usage of words in the human text often complies the Zipfian law, which indicates the topmost frequent words double the second, three times the third, etc. Nguyen-Son et al. [8] use this law for detecting machine translated document. Furthermore, they extracted useful humanity text including idiom, cliché, ancient, and dialect phrases. They also estimated the relationships among certain phrases based on co-reference resolutions. These features only work well on a large text in which the word distribution is more stable and additional features appear more.

2.1.4 Coherence

Although the machine-translated text can preserve the meaning, the coherence of such text is still low. Some researchers have measured the coherence to distinguish the machine text with the human text. For example, Nguyen-Son et al. [9] matched similar words between two sentences in a paragraph. The similarity between two matched words is used to estimate the coherence. In another work, Nguyen-Son et al. [10] broadened the matching on any words in the paragraph in both within and across sentences. However, the coherence is tight in a paragraph but is downgraded in other levels such as sentence and document.

2.2 Other Machine-Generated Text Detection

Many other machine-generated texts support for malicious purposes such as paper generation and fake review. Labbé and Labbé [12] prove that artificial papers are produced by using abundant duplicated words and phrases. Therefore, they suggested an inter-textual distance to estimate the similarity between two word distributions and used the distance to recognize the machine-generated text. In fake review detection, Juuti et al. [11] extracted features from thirteen readability metrics. Moreover, they used N-gram models for various text components including words, simple POS, detailed POS, and syntactic dependency. The duplicated usages of word distribution and N-gram model indicate high relevant between machine-translated and other machine-generated texts detection.

2.3 Deep Learning Methods

Deep learning-based methods for detection have been used to increase the accuracy of many natural language processing tasks; such as paraphrase detection. Paraphrasing is another method through which adversaries can carry out

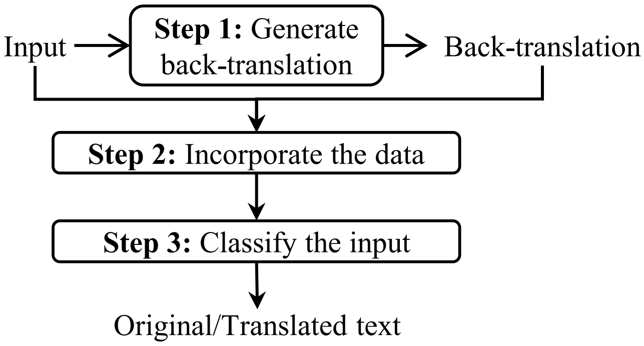
Input
Step 1: Generate back-translation
Back-translation
Step 2: Incorporate the data
Step 3: Classify the input
Original/Translated text

Fig. 3 The proposed schema for detecting machine-translated text.

plagiarism, moreover, the task requires us to distinguish between two very similar sentences. Hence due to the lack of previous research on using deep learning networks for detecting machine-translated text, we can consider the next closest task- paraphrase detection for comparison. Existing methods suggest using convolutional neural networks (CNN) and long short term memory (LSTM) networks [15]. For the purpose of our experiments we mainly assessed two models, CNN for sentence classification [16] and bidirectional encoder representations from transformers (BERT) [17]. The CNN model achieves good classification performance across a wide range of text classification tasks like sentiment analysis making it a standard baseline for new text classification architectures. It is able to learn word meaning and relations but lacks the ability to understand the context. The BERT network applies the bidirectional training of the transformer to an attention model for language modeling. BERT achieved some of the highest results on multiple NLP tasks including paraphrase detection on Microsoft paraphrase corpus (MRPC), question answering (SQuAD v1.1), natural language inference (MNLI), and others making it the current state-of-art model for NLP. This model performs well on sentence pair tasks as it uses the attention mechanism to generate deep context-based embeddings making it ideal for identifying relationships between sentence pairs. For the purpose of experimenting with deep learning methods, we assessed these models on a dataset with machine-translated text and human text (without integrating back-translation features)

3. Proposed Method

The schema of the proposed method includes three steps as shown in Figure 3:

- **Step 1 (*Generate back-translation*)**: Machine Translator (e.g., Google) is used to generate the back-translation of the input text.
- **Step 2 (*Incorporate the data*)**: The input text and its back translation are combined together to generate a sentence pair.
- **Step 3 (*Classifying the input*)**: The BERT model is used to classify the sentence pair, learning similarity features between the text and its back-translation.

The following subsections describe the step-by-step of the proposed method.

3.1 Generating Back-Translation (Step 1)

The input text in the original language is translated into an intermediate language, which is different from the original one. The translated version is then re-translated back to the original language. The final translation is called as back-translation. In this paper, we use Google as a translator. In Figure 2, the back-translations h_2 and m_3 are generated from the human text h_0 and machine text m_1 respectively with the intermediate language, French.

Figure 4 shows an example of back-translation detection. In particular, we use Japanese for generating the machine-translated text m'_2 from the original text m'_0. For distinguishing the two input texts h'_0 and m'_2, we create their back-translated texts h'_2 and m'_4 respectively with Japanese. Like Figure 2, we highlight the variants between the input texts and their back-translation with bold for word usages and underline for structures. h'_2 makes more variants than m'_4. Again, the translation with four times in m'_4 causes fewer changes than that with two times in h'_2.

3.2 Incorporate the Data (Step 2)

This step aims to create a combined data set between the input text and the generated back-translation. Due to the relationship between back-translated text and machine or human-generated text, we want to create a data set which can exploit this relationship. So for each of the inputs, we parallelly add its machine-generated back-translation. Then the dataset is labeled according to the original labels of the input text. Finally, we divide the dataset for training and evaluation. In the case of a parallel corpus (eg. EuroParl), we also ensure that both the machine-translated text (from parallel French translation) and human text (original English text) are present in the same group as this will help the detector learn to identify the difference between the two types more easily. This will help in creating a balanced dataset which is important for improving the performance of our method.

3.3 Classifying the Input (Step 3)

The combined data set from the previous step is divided into the training set and evaluation set. We train the BERT model on the sentence pair classification task on the combined dataset with each input text and corresponding back-translation forming the sentence pair. We recommend using the BERT base model with the following hyperparameter values for optimized accuracy: maximum sequence length as 128, batch size 32, learning rate 2e-5 and number of epochs as 3. Finally, we evaluate the trained model on the evaluation set and classify the input text is translated by a machine or written by a human. The BERT model is unique from other models as it is good at identifying feature similarities between sentence pairs. Hence making BERT the ideal choice for our proposed method.

	m'_0(English)=h'_0: "*One of the best examples of how to treat a subject, you're not fully aware is being examined, much like a photo of yourself you didn't know was being taken.*"
	m'_1(Japanese): "*被験者の治療方法の最良の例の1つは、自分が知らなかった自分の写真が撮影されているように、検査されていることを完全に認識していないことです。*"
h'_0(English)=m'_0: "*One of the best **examples of how** to treat a subject**, you're** not fully aware **is** being examined, **much like** a **photo** of **yourself you didn't** know **was being taken.***"	**m'_2(English):** "*One of the best ways to treat a subject is that **they are not fully aware** that **they** are being examined, as if **they had taken** a picture of **themselves** that they did not know.*"
h'_1(Japanese): "*被験者の治療方法の最良の例の1つは、自分が知らなかった自分の写真が撮影されているように、検査されていることを完全に認識していないことです。*"	**m'_3(Japanese)**: "*被験者を治療する最良の方法の1つは、自分が知らない自分の写真を撮ったかのように、検査されていることを完全に認識していないことです。*"
h'_2(English): "*One of the best **ways** to treat a subject **is that they are** not fully aware **that they are** being examined, **as if they had taken** a **picture** of **themselves that they did not** know.*"	**m'_4(English):** "*One of the best ways to treat a subject is **to be completely unaware** that **you** are being examined, as if **you took** a picture of **yourself** that **you** did not know.*"

Fig. 4 Human vs machine text in back-translation detection.

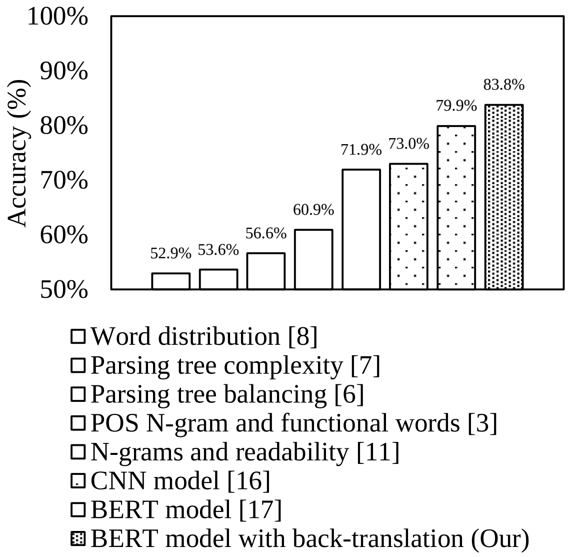

Fig. 5 Translation detection results.

4. Evaluation

4.1 Translation Detection

4.1.1 Dataset

We randomly selected 45000 English-French sentence pairs from the European parallel corpus[*3]. This dataset consists of records of statements given in the European parliament with a human translation in languages like French, Swedish, and Spanish. While the English was used as human-written texts, the human translated French text was translated to English by Google Translator for producing machine-translated texts. The sentences are merged together to create a dataset of 90000; the integrated dataset contains 25.3 words per sentence on average. We then split the dataset into two parts: 86000 sentences for the train set and the remaining for the test set. To balance between human and machine texts in each set, we distributed both human and corresponding machine sentences into the same set.

4.1.2 Comparison

First we conducted experiments with the previous feature-based methods like parsing tree [6], [7], N-gram model [3], [11], word distribution [8], and deep-learning-based method including CNN-model [16] and the BERT network [17]. As we needed to choose an evaluation metric which would be common for all of these methods, we have used accuracy to create a comparison between their results. The bar graph chart 6 shows the comparison between the feature-based methods in the first five columns filled with no pattern. The next two columns show the results of deep learning models like CNN and BERT network, filled with a light pattern. The last column with the darkest pattern is the result of our proposed method with the BERT model. For this experiment, we had chosen French as our language for the detector i.e used French as the intermediate language to produce the back-translation.

One clear trend we can see is that deep-learning-based methods have higher accuracy than feature-based methods. This indicates that we are able to extract more information about context and words by using word embedding like Word2vec [18] and BERT embedding [17] than the features used in methods like N-gram model [3], [11]. This helps

[*3] `https://statmt.org/europarl/`

in creating a more meaningful distinction between machine-translated text and human text based on their word usage and syntactic coherence, explaining the difference of accuracy between deep learning-based methods and feature-based methods. With the highest accuracy of feature-based method-N gram and readability [11] being 71.9% when compared to the accuracy of the BERT model [17] as 79.9%.

The last column gives the accuracy of our suggested method. For this experiment, we followed the schema by integrating back-translation text of our previous dataset along with the input and finally into the BERT model. This gave an accuracy of 83.8% which beats the highest accuracy of all previous methods by a margin of 3.9%. It demonstrates that our method of using back-translation information improves performance. So the only use of internal text information is insufficient to recognize machine-translated texts.

4.2 Back-Translation Detection

4.2.1 Dataset

We also checked the capability of our proposed method on another task, namely back-translation detection. Back-translation can be easily used to generating paraphrasing texts for supporting malicious purposes such as fake reviews, political posts, and plagiarism. The generation needs only an original text and using back-translation with various languages for generating many paraphrasing versions. For simulating this scenario, we picked up all 11748 sentiment sentences from Stanford Treebank corpus[*4]. The data include both positive and negative sentiment sentences. Then we generated the back-translated texts which are considered as machine sentences. For this experiment, we also wanted a comparison between using different languages for generating and detecting so we used two different languages for creating the dataset. We chose French as our rich resource language and Japanese as our low resource language. The machine sentences were integrated with the original ones into 23496 sentence dataset that averages at 19.1 words per sentence. It is obviously smaller than the translation dataset above due to short sentences such as "*Imperfect?*" and "*Cool.*" The dataset was also split into train and test sets with balancing human and machine sentences in the same manner with the section 4.1.1. With 20000 sentences in the training set and 3496 sentences in the test set.

4.2.2 Comparison

We conducted similar experiments with the previous methods on this back-translation dataset. For evaluating our detectors, we use French for the rich resource dataset. The results are shown in the left graph of Figure 6.

The performances of all the methods on are greater than the performance in comparison to translation-detection experiment. With French accuracy of back translation dataset being 87.1% as compared to 83.8% of the translation dataset. The main reason is that the back-translation machine texts are generated after using the translator in two times. Therefore, the quality is downgraded and this text is more easily distinguishable. The trend followed is similar to the translation-detection dataset with deep learning methods having a higher accuracy than feature-based methods for the same reasons as mentioned in the section 4.1.2.

4.2.3 Low-resource language

We examined similar experiments with low-resource languages. With the same dataset of 11748 human sentiment sentences, we choose Japanese for generating machine back-translated texts. For detectors, we used the same intermediate language Japanese. The results are shown in the right graph of the Figure 6.

In the previous methods, the results are quite similar to detecting back-translation with the rich-resource language. Comparing the results from rich resource and low resource languages with our method, we see that the results of Japanese dataset are 89.9% and French dataset are 87.1%. This can be attributed to the quality of translation generated via Japanese and French. Using Japanese, the quality of translation is low and so the back-translation can be easily distinguished from its human text counterpart. This also shows the persistence of our method with different languages with both low and high-quality translations. Our experiments outperform the state of the art method in both Japanese and French datasets by 3.7% and 4.9% respectively.

5. Conclusion

In this paper, we have exploited that when using machine translators many times, the translated text converges. Moreover, the variant between two consecutive usages gets to be smaller. By exploiting this property, we then propose a method for detecting two types machine-generated text: machine translation and machine back-translation. In machine translation detection, the evaluation of French sentences on previous methods shows that our method can detect translated text with 83.8% accuracy. It outperforms the previous methods with the best accuracy as 79.9%. In back-translated text detection, the performance is even significantly improved from 82.2% to 87.1% for rich resource language and from 86.2% to 89.9% for low resource language. This shows the robustness of our method in detecting back-translation generated from different languages. Moreover, it proves that the integration of back-translation information improves classification accuracy. We show that through our methods we can achieve an average increase of 4.1% in accuracy over the best previous methods.

In future work, we will investigate the effect of our findings on different languages such as Swedish and Spanish, to study the effects of different language combinations. We will also try and increase the accuracy of our proposed method by integrating BLEU features with the contextual BERT embeddings to form a hybrid model. Moreover, we will also conduct experiments on different data types like computer-generated fake news, fake reviews and different kinds of adversarial texts. Beyond text, the applications using our hy-

[*4] `http://nlp.stanford.edu/~socherr/stanfordSentimentTreebank.zip`

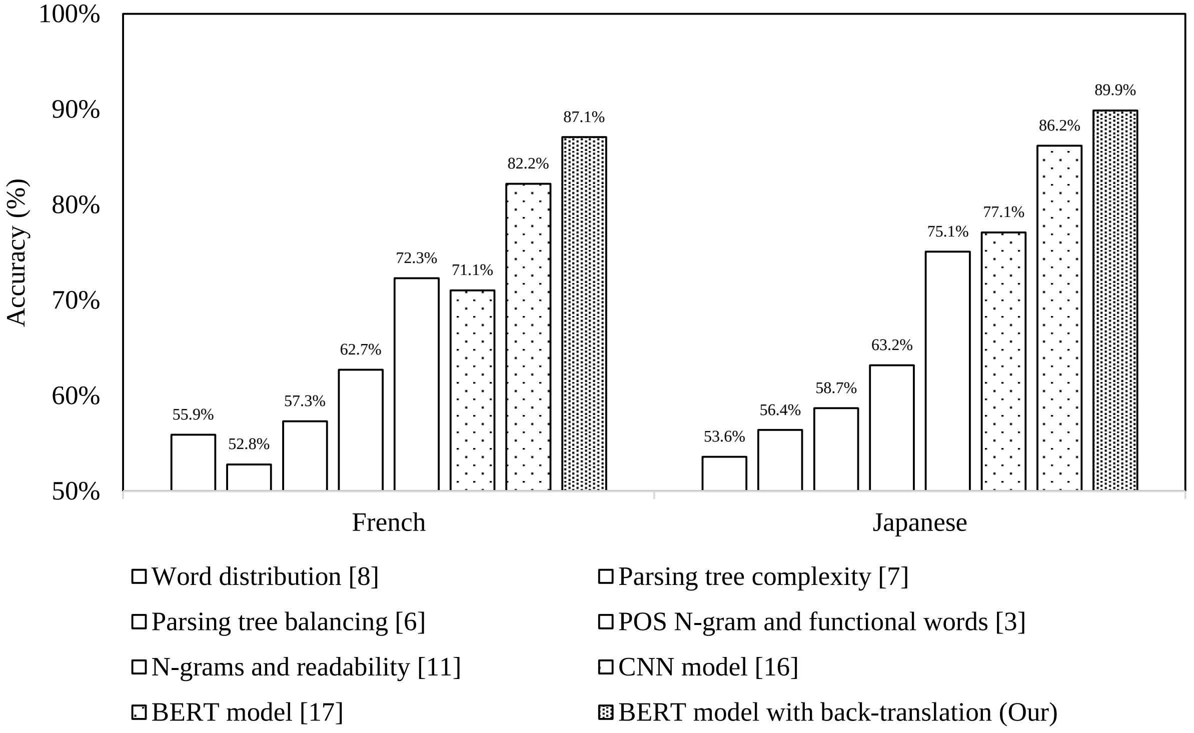

Fig. 6 Back translation detection results.

pothesis for detecting other machine-generated data (e.g., image, video, sound, and structured data) will also be considered.

References

[1] M. Jones and L. Sheridan, "Back translation: an emerging sophisticated cyber strategy to subvert advances in digital ageplagiarism detection and prevention," *Assessment and Evaluation in Higher Education*, vol. 40, no. 5, pp. 712–724, 2015.

[2] S. Prabhumoye, Y. Tsvetkov, R. Salakhutdinov, and A. W. Black, "Style transfer through back-translation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 866–876.

[3] R. Aharoni, M. Koppel, and Y. Goldberg, "Automatic detection of machine translated text and translation quality estimation," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2014, pp. 289–295.

[4] Y. Arase and M. Zhou, "Machine translation detection from monolingual web-text," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2013, pp. 1597–1607.

[5] H.-Q. Nguyen-Son and I. Echizen, "Detecting computer-generated text using fluency and noise features," in *Proceedings of the International Conference of the Pacific Association for Computational Linguistics (PACLING)*, 2017, pp. 288–300.

[6] Y. Li, R. Wang, and H. Zhao, "A machine learning method to distinguish machine translation from human translation," in *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation (PACLIC)*, 2015, pp. 354–360.

[7] J. Chae and A. Nenkova, "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 Association for Computational Linguistics (EACL)*. Association for Computational Linguistics, 2009, pp. 139–147.

[8] H.-Q. Nguyen-Son, N.-D. T. Tieu, H. H. Nguyen, J. Yamagishi, and I. Echizen, "Identifying computer-generated text using statistical analysis," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2017, pp. 1504–1511.

[9] ——, "Identifying computer-translated paragraphs using coherence features," *ArXiv Preprint arXiv:1812.10896*, 2018.

[10] H.-Q. Nguyen-Son, T. P. Thao, S. Hidano, and S. Kiyomoto, "Detecting machine-translated paragraphs by matching similar words," in *ArXiv Preprint arXiv:1904.10641*, 2019.

[11] M. Juuti, B. Sun, T. Mori, and N. Asokan, "Stay on-topic: Generating context-specific fake restaurant reviews," in *Proceedings of the European Symposium on Research in Computer Security (ESORICS)*, 2018, pp. 132–151.

[12] C. Labbé and D. Labbé, "Duplicate and fake publications in the scientific literature: how many scigen papers in computer science?" *Scientometrics*, vol. 94, no. 1, pp. 379–396, 2013.

[13] H. Somers, F. Gaspari, and A. Niño, "Detecting inappropriate use of free online machine translation by language students–a special case of plagiarism detection," in *Proceedings of 11th Annual Conference of the European Association for Machine Translation (EAMT)*, 2006, pp. 41–48.

[14] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," in *Proceedings of the 2018 the Conference on Empirical Methods in Natural Language Processing (EMNLP) Workshop*, 2018, pp. 353–355.

[15] B. Agarwal, H. Ramampiaro, H. Langseth, and M. Ruocco, "A deep network model for paraphrase detection in short text messages," *Information Processing & Management*, vol. 54, no. 6, pp. 922–937, 2018.

[16] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1746–1751.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 2019, pp. 4171–4186.

[18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of International Conference on Learning Representations (ICRL)*, 2013.