IberLEF 2023 AuTexTification: Automated Text Identification Shared Task – Team OD-21

Rinaldo Gagiano[1,2,*], Haytham Fayek[1], Maria Myung-Hee Kim[2], Jennifer Biggs[2] and Xiuzhen Zhang[1]

[1]School of Computing Technologies, RMIT University, Melbourne, Australia
[2]Defence Science and Technology Group, Australia

Abstract

The emergence of large Pre-trained Language Models (PLMs) has revolutionized the generation of human-like text, with implications spanning various domains. However, this progress also brings challenges, including the proliferation of machine-generated text that can be misleading or malicious. To address this, the detection of machine-generated text has become crucial. In this paper, we participate in the IberLEF 2023 AuTexTification shared task, focusing on binary classification of machine-generated versus human text and authorship attribution across English and Spanish. We employ a deep learning approach using transformer-based PLMs and data augmentation techniques in an attempt to enhance model performance. Our results for the shared task yield an average macro F1-score of 63.02 for Subtask 1 and 58.39 for Subtask 2, showcasing our competitive performance with best run ranks of 11 out of 52 and 10 out of 38, respectively.

Keywords

Shared task, Artificial text detection, Machine-generated text classification, Pre-trained language models, Neural authorship attribution, Transformer-based attribution

1. Introduction

Publicly available large Pre-trained Language Models (PLMs) such as ChatGPT [1], LLaMA [2], and Bard [3], possess the remarkable ability to generate text that closely resembles human-created content in terms of coherence, style, and grammar [4]. This remarkable capability has led to the emergence of various beneficial applications across diverse domains, including healthcare [5, 6, 7], legal [8, 9] and finance [10, 11, 12]. However, alongside the positive applications, it is essential to acknowledge the potentially darker side associated with PLM exploitation. PLMs have been used to disseminate fabricated news stories [13, 14, 15], manipulate public opinion [16, 17, 18], and carry out academic fraud [19, 20, 21]. Furthermore, it has been utilised to generate content that is discriminatory or otherwise offensive [22, 23, 17], with the potential to harm individuals or communities.

IberLEF 2023, September 2023, Jaén, Spain
*Corresponding author.
rinaldo.gagiano@student.rmit.edu.au (R. Gagiano); haytham.fayek@ieee.org (H. Fayek); maria.kim@dst.defence.gov.au (M. M. Kim); jennifer.biggs@dst.defence.gov.au (J. Biggs); xiuzhen.zhang@rmit.edu.au (X. Zhang)
https://www.rinaldogagiano.com/ (R. Gagiano)

The generation of malicious or influential content, pervasive across the internet [24] and at scale, represents a significant concern in this era of advanced language models. Therefore, it is imperative to address the ethical implications associated with it and implement appropriate safeguards to mitigate the risks arising from the misuse of these models. Developing technology that can automatically detect machine-generated text becomes crucial in tackling these challenges. Such advancements would serve as a valuable tool in safeguarding against the negative impacts of misleading or harmful content. These automated detection systems could aid in content moderation and support protective measures aimed at maintaining the integrity and safety of online platforms.

The field of detecting machine-generated text has seen significant advancements with the use of deep learning models such as BERT [25], GROVER [15], and RoBERTa [26]. These models have shown promising results and have become state-of-the-art (SOTA) approaches in this domain [27, 28, 29, 30]. As the interest in AI-generated text detection grows, so too does the activity in "AI-generated" text detection with organisations offering API based zero-shot machine-generated text detectors such as Turnitin [31], GPTZero [32], ZeroGPT [33], OpenAI [34], and GPTRadar [35]. In more complex detection scenarios, such as authorship attribution, PLMs have demonstrated their potential effectiveness. They outperform traditional stylometric classifiers and have become the go-to choice for this task [36, 37]. However, it is important to note that PLMs face challenges in multi-class classification tasks compared to binary classification tasks [27, 37]. Neural authorship attribution, which involves identifying the specific generative model used, can be particularly difficult for PLMs. Moreover, older PLMs that performed well in earlier classification tasks [27, 38, 39] may struggle to detect text generated from more recent models [40, 37]. The evolving nature of generative models and their capabilities require continuous adaptation and improvement of detection techniques. Overall, the field of detecting machine-generated text is evolving rapidly, with ongoing research and development focused on enhancing the performance and robustness of detection models to keep up with the advancements in AI-generated content.

In this paper, we participate in the IberLEF 2023[41] AuTexTification shared task[42], which focuses on automated text identification. The task consists of two subtasks: (i) detecting machine-generated text by classifying samples as either "generated" or "human," and (ii) authorship attribution, where the goal is to identify the source of a sample among six different options. These subtasks are available in both English and Spanish. To tackle these challenges, we adopt a deep learning approach and leverage various PLMs in a controlled training environment. We aim to improve model performance by exploring data augmentation techniques. Specifically, we investigate the impact of dataset length subsetting, where we create subsets of shorter and longer samples, as well as dataset translation concatenation, where we translate samples between languages and merge them with the original dataset. Through systematic training and evaluation, we rank the performance of different models based on the macro-F1 score obtained during validation. By participating in the shared task and conducting these experiments, we aim to contribute to the advancement of automated text identification techniques and provide valuable insights into the performance of different models and data augmentation strategies.

For Subtask 1, our average macro-F1 score is 63.02, and the best run achieves a rank of 11 out of 52 for the Spanish portion. In Subtask 2, our average macro-F1 score is 58.39, and the best run attains a rank of 10 out of 38 for the English portion. These results indicate the performance

of our models in the task, with higher scores indicating better performance. We compare our performance against other participants in terms of rank, showcasing our relative standing in the competition.

2. Related Work

Text classification, which involves extracting features from raw text data and predicting the categories of text based on these features, has been a topic of extensive research. Over the past few decades, various models have been proposed to address this task.

Among the traditional models, the Naive Bayes model [43] stands out as one of the earliest approaches used for text classification. Subsequently, other generic classification models such as Logistic Regression [44], Support Vector Machines [45], Random Forest [46], and K-Nearest Neighbours [47] have been widely assessed [48, 49] as classifiers in text classification tasks.

Over time, more advanced machine learning boosting techniques have emerged as powerful tools for text classification. Models such as Extreme Gradient Boosting [50] and Adaptive Boosting [51] have gained attention due to their potential to deliver excellent performance. These boosting algorithms have been successfully applied to text classification problems [52, 53, 54, 55, 56], demonstrating their effectiveness.

The use of deep learning models marks a turn in the approach to the text classification task. Classification models based on Convolutional Neural Networks [57] and Recurrent Neural Networks (RNN) [58] have been shown to outperform more traditional methods [59, 60, 61] and have garnered substantial attention in the context of text classification.

More recently, Transformer-based language models have become a popular choice in developing text classification models due to their remarkable performance on various text classification datasets. Since the introduction of the Transformer architecture, Transformer-based models have emerged as the SOTA in text classification. These models have revolutionized the way we approach text classification by harnessing the power of self-attention mechanisms [62] and offering increased parallelization capabilities [63] compared to traditional models like RNNs.

Transformer-based models, such as GPT-2 (Generative Pre-trained Transformer) [64], BERT (Bidirectional Encoder Representation from Transformers) [25], GROVER (Generating aRticles by Only Viewing mEtadata Records) [15], and RoBERTa (Robustly Optimized BERT Approach) [26], have achieved remarkable performance across a wide range of NLP benchmarks, including text classification [65, 39, 66, 67]. These models have demonstrated their ability to capture nuanced contextual information, resulting in improved accuracy and robustness in classifying text data.

The surge of interest in Transformer-based models has sparked the development of various model variants that aim to enhance the state-of-the-art performance in Natural Language Processing tasks. DistillBERT [68], a derivative of BERT, utilises knowledge distillation during pre-training to compress the size of the BERT model while preserving its original capabilities. ERNIE 2.0 [69] incorporates domain-specific knowledge from external knowledge bases, such as named entities, into the model's pre-training process. XLNet [70] adopts a unique permutation-based training approach, enabling it to capture dependencies between all positions in a sequence, leading to an improved understanding of language context. XLM-RoBERTa, built

upon the RoBERTa architecture, is a cross-lingual language model that is designed to effectively model multilingual text by pre-training on large-scale datasets from multiple languages. BLOOM [71] is a specialized decoder-only Transformer language model that has been trained on a diverse dataset encompassing 46 natural languages and 13 programming languages. These advancements highlight the continuous innovation in Transformer-based models, pushing the boundaries of NLP performance. Based on these identified models, in this paper, we aim to compare the effectiveness of these models and explore the impact of data augmentation techniques on improving their performance.

3. Methodology

We adopted a strategy that revolved around leveraging PLMs for the AuTextification shared task. We explored various approaches such as data augmentation techniques to improve model performance and a systematic evaluation of a select number of PLMs.

3.1. Constraints

The task allows the use of publicly pre-trained language models. Only the text derived from the training data is allowed to be used. No external data can be used for further model pre-training. Usage of data from one subtask in the other subtask is not allowed.

3.2. Dataset

3.2.1. Data Description

Subtask 1 is a binary classification problem with target labels 'human' and 'generated'. The text style for this subtask covers five different domains, where three domains (legal documents, how-to articles, and social media) will be used in the training set and two hidden domains (news, reviews) will be used in making up the test set.

Subtask 2 is a multi-class classification problem with target labels 'A', 'B', 'C', 'D', 'E', and 'F', indicating the generation model used. Generative models range in size from 2 billion parameters to 175 billion parameters and correlate to these labels: *{"A": "bloom-1b7", "B": "bloom-3b", "C": "bloom-7b1", "D": "babbage", "E": "curie", "F": "text-davinci-003"}*.

The identity of hidden domains and generation models was only revealed after the task submission deadline.

3.2.2. Data Distribution

To assess the distribution of the training samples, we calculate the character length and token length of each sample in the provided training datasets. We tokenize the data using a space separation method. We average these lengths, per task, as shown in Table 1.

Comparing the statistics of the training datasets, we observe that the distribution of samples and average lengths are similar within subtasks. It is worth noting that, on average, Subtask 1 has a smaller number of samples compared to Subtask 2, with 32,954 and 22,176 samples, respectively. Target labels are approximately even across all tasks.

Table 1
General statistics of training datasets. (EN): English, (ES): Spanish, Char: Character.

Task	Samples	Char Average	Token Average
Subtask 1 (EN)	33845	305	54
Subtask 1 (ES)	32062	307	52
Subtask 2 (EN)	22416	335	60
Subtask 2 (ES)	21935	340	58
Average	**27564**	**322**	**56**

The range of sample token lengths for both subtasks is 1 to 131 tokens long. For Subtask 1, 35.8% and 45.76% of samples are between 15 to 30 tokens and 65 to 85 tokens long, respectively. This distribution suggests the presence of two primary sample formats within the data. We hypothesise that samples from the social media domain fall into the shorter length category, while samples from legal documents or how-to articles domains would fall into the longer length category. For Subtask 2, we see a similar bimodal distribution with 21.09% and 51.12% of samples being found between 20 to 25 tokens and 70 to 90 tokens in length, respectively. This could identify a similar domain split as highlighted in Subtask 1.

3.3. Data Pre-processing

For each subtask, the training data is split into training and validation sets, divided into 80% and 20% respectively of the original data. Dataset splitting is stratified on the target labels. The *DataCollatorWithPadding* function from the Transformers package is used to pad all batches to the same length. We use the following label encodings: *{"generated": 0, "human": 1}* and *{"A": 0, "B": 1, "C": 2, "D": 3, "E": 4, "F": 5}*.

3.4. Training Scenario

To gain an understanding of the capabilities and shortcomings of current SOTA PLMs, we utilised a large variety of models for both subtasks. To ensure results were comparable, the same training parameters were used for each model and experiment. The training was conducted across eight Tesla V100 GPUs. The batch size was set to 16 for both training and validation. AdamW[72] was used as the optimiser. Both optimiser and unspecified training parameters were left as default [73]. We ran model training for 5 epochs on Subtask 1. We increased this to 15 epochs for Subtask 2, due to the limited number of samples and increased task complexity.

The models used are as follows: BERT (Base uncased, Large uncased), BERTIN-RoBERTa, BLOOM, DistilBERT (Base uncased, Base cased, Multilingual cased, Spanish uncased), ERNIE 2.0, GPT-2 (Small, Medium), RoBERTa (Small, Large), XLM-RoBERTa, and XLNet. All models, along with model cards and parameter sizes can be found on Hugging Face [74].

3.5. Evaluation Scheme

The macro-F1 score is the metric used for shared task evaluation and ranking. For consistency, we conduct all experiment evaluations using the macro-F1 as our metric.

4. Experiments

We carry out three experiments to examine the performance of PLMs on the shared task. These experiments aim to investigate the impact of data augmentation techniques and the effectiveness of different PLMs on overall model performance. Specifically, the first two experiments focus on data augmentation. We manipulate the training dataset to assess its influence on model performance. By varying the composition of the dataset, we aim to understand how these changes affect the models' ability to perform well on the task. In the third experiment, we systematically evaluate the performance of multiple PLMs across different subtasks of the shared task. By comparing the performance of various PLMs, we can identify which models are more effective in handling the specific challenges posed by the shared task.

4.1. Importance of Text Length Variation on Model Performance

In this first experiment, we aim to investigate the impact of text length on classification performance. Previous studies have shown that models tend to classify longer texts more accurately compared to shorter texts [75, 76, 77]. Based on this observation, we hypothesised that the presence of shorter samples in our training set might introduce noise during the training process and potentially hinder overall performance.

To test this hypothesis, we divided the Subtask 1 (EN) training set into two subsets based on token length. The first subset, referred to as *Short* subset, consisted of samples with a total token length shorter than the sample average of 55. The remaining samples were included in the *Long* subset. The original dataset is denoted as *All*.

To evaluate the performance of the subsets, we trained a GPT-2 (Medium) model on each dataset. We extract the highest macro-F1 score on the validation set across epochs as a measure of performance. The validation set contains both short and long samples. Results can be seen in Table 2.

Table 2
Validation macro-F1 scores for GPT-2 (Medium) trained on *All*, *Short*, and *Long* subsets.

Dataset	Macro-F1
All	**0.895**
Long	0.849
Short	0.684

The performance of the *Short* subset (Table 2) demonstrates that training solely on short samples does not generalize well to longer samples that are unseen during training. In contrast, the performance of the *Long* subset remains relatively good. This suggests that learning from long samples alone can still yield reasonable results.

The best performance is achieved when the model is trained on both short and long samples, as demonstrated by the *All* dataset (Table 2). The variation in sample lengths helps the language model to develop a more diverse and robust pattern recognition capability. Overall, the findings strongly indicate that including a mixture of short and long samples in the training dataset is ideal for maximizing the performance of the language model.

4.2. Data Augmentation through Machine Translation

In this experiment, we aim to explore the effectiveness of data augmentation through translation. Given that Subtask 2 has a smaller training set sample size and increased task complexity compared to Subtask 1, we hypothesised that increasing the sample size would lead to improved overall model performance during training. To increase the sample size while staying within the constraints of the shared task, we employed PLMs to translate our Spanish dataset into English and vice versa. Considering all samples in Subtask 2 are authored by machines, the translation done by another machine would not compromise the source as it would in Subtask 1, therefore we only utilise Subtask 2 for this experiment.

The translation of samples was performed using two translation PLMs. We select the opus-mt-en-es [78] model for translating from English to Spanish, and the opus-mt-es-en [79] model for translating from Spanish to English. These translated datasets were then concatenated with their respective target language datasets. We named the original datasets as *Base* and the translated datasets as *Trans*.

To assess the impact of these crafted datasets, we trained three different PLMs (DistilBERT (Multilingual cased), DistilBERT (Base uncased), and RoBERTA (Small)) on each dataset and calculated the average macro-F1 score across the models on the validation set. The validation set consisted solely of samples from the respective *Base* samples. Results are listed in Table 3.

Table 3
Averaged validation macro-F1 scores for models trained on *Base* and *Trans* datasets.

Dataset	Macro-F1
English Base	0.571
English Trans	0.556
Spanish Base	0.587
Spanish Trans	0.564

Analysis of model performance reveals that including translated samples in the training dataset hinders overall performance (Table 2). The English *Trans* and Spanish *Trans* datasets performed 0.015 and 0.023 worse, respectively, compared to their *Base* counterparts. One possible reason for this outcome is the introduction of noisy samples through the translation process. Translated samples may contain improper translations in terms of grammar or coherency, which can negatively impact the model's ability to learn and generalize effectively to cleaner test samples. Further, the translation conducted by a single model may remove original model artifacts and induce its own artifacts, causing model performance degradation.

4.3. Systematical Comparison of SOTA PLMs

In the absence of data augmentation techniques yielding improved performance scores, our focus shifts to comparing different models under the same training confines. Each experiment involves training models on a specific subtask, where the highest macro-F1 score is recorded. The experimental results presented in Table 4 demonstrate results for each model and its performance within that respective subtask.

Table 4
Validation macro-F1 scores.
Bold: Best Macro-F1, Underline: 2nd Best Macro-F1, *: 3rd Best Macro-F1, ST: Subtask, EN: English, ES: Spanish.

Model	ST1 (EN)	ST1 (ES)	ST2 (EN)	ST2 (ES)
bert-base-uncased	0.928	0.897	0.575	0.579
bert-large-uncased	0.928	0.894	-	0.589
bertin-roberta	-	-	-	0.586
bloom	0.917	0.890	-	-
distilbert-base-cased	0.919	0.886	0.576	0.567
distilbert-base-multilingual-cased	0.924	0.920	0.570	**0.600**
distilbert-base-spanish-uncased	-	-	-	0.592
distilbert-base-uncased	0.923	0.894	0.563	0.574
ernie-2.0-large-en	**0.943**	**0.922**	**0.597**	0.577
gpt2-medium	0.936*	0.898	-	-
gpt2-small	0.902	0.879	-	-
roberta-large	0.933	0.337	0.588*	0.588
roberta-small	0.937	0.916*	0.578	0.592*
xlm-roberta-large	0.335	0.337	0.595	-
xlnet-large-cased	0.923	-	-	-

Findings (table 4) indicate that the ERNIE 2.0 model achieved the highest average macro-F1 score across both languages for Subtask 1 and the English version of Subtask 2 with 0.943, 0.922, and 0.597 macro-F1 scores, respectively. For Subtask 2 (ES), the Multilingual DistilBERT (Base cased) model exhibited the best performance, achieving the highest macro-F1 score of 0.6.

The lower scores achieved in Subtask 2, compared to Subtask 1, indicate an increase in task difficulty from binary to multi-class classification. Most models, even after numerous epochs, only marginally outperform chance-level performance (0.5). This further highlights the complexity and challenges associated with multi-class classification tasks. It is also worth noting that all models achieved similarly low macro-F1 scores for Subtask 2, suggesting that the model performance in this experimental setup may be more influenced by the initial seeding and optimiser biases [80], rather than the underlying model architecture and training. Therefore, re-running the experiments for both languages in Subtask 2 might yield similar scores but different rankings amongst the models.

Interestingly, the XLM-RoBERTa model exhibited substantially lower performance across Subtask 1, which could be attributed to a potential training error rather than inherent difficulty with the task. Additionally, it seems the RoBERTa (Large) model also experienced a similar issue with Subtask 1 (ES). Due to time constraints, the BERTIN-RoBERTa and Spanish DistilBERT models were only used for Subtask 2 (ES). We do not assess BLOOM, and either GPT-2 models on Subtask 2. XLNet's missing performance for all but one subtask was due to resource issues.

5. AuTexTification Shared Task Submission

Participants are allowed to submit three runs per language for each subtask. To generate predictions on the test samples, we select the top three models based on their macro-F1 scores from our experiments (Section 4.3). From this selection, we retrain the models and save the top three checkpoints for each model, based on the macro-F1. As a result, we end up with a total of nine checkpoints per task.

5.1. Run Composition

For each run in our submission, we employed different strategies to leverage the available model checkpoints. Run strategy is conducted for each subtask.

Run 1: The checkpoint with the highest recorded macro-F1 score and associated model is used to generate predictions.

Run 2: The top three checkpoints with the highest recorded macro-F1 score and associated models are chosen to produce classification logits. We then summed these logits and selected the index with the highest logit value as the prediction label. By including multiple models in this way, we aimed to induce some variation in the predictions.

Run 3: From the nine model checkpoints available, we randomly selected three checkpoints. After producing and summing the classification logits from these checkpoints, we extracted the index with the highest logit value as the prediction label. This method was chosen to introduce even more variation in the predictions compared to the previous run methods.

6. AuTexTification Shared Task Results and Discussion

Displayed in Table 5 are the official results of our best runs per task along with their associated macro-F1 scores, overall run rank, and overall team rank.

Table 5
Team OD-21 best run results for AuTexTification shared task. EN: English, ES: Spanish, *: Includes baselines in ranking.

Task	Best Run #	Macro-F1 Score	Run Rank*	Team Rank*
Subtask 1 (EN)	Run 2	60.33	34/76	21/41
Subtask 1 (ES)	Run 2	65.71	11/52	9/28
Subtask 2 (EN)	Run 1	58.38	10/38	6/24
Subtask 2 (ES)	Run 3	58.39	14/29	6/19

The macro-F1 scores for Subtask 1 exhibited a notable decline in comparison to the experiment scores, which can be attributed to the presence of out-of-distribution samples caused by domain shift. Specifically, the domains of the test set (news and reviews) differed from those encountered during training (legal documents, how-to articles, and social media). This domain shift causes a distributional difference which compounded by the limited size of the training sample, has been shown to restrict transformer-based models' capacity to effectively capture generalisable features [81] of out-of-distribution samples.

The macro-F1 scores for Subtask 2 were on par with our experiment results. However, this may be attributed more to chance rather than actual model performance, as both experiment and test scores remained relatively low and are typical indications of under-fitting. This observation is exemplified by our Subtask 2 (ES) best run, as this run had the most induced prediction variation among the different run methods.

7. Conclusion

In this paper, we participated in the IberLEF 2023 AuTexTification shared task, focusing on automated text identification and authorship attribution in English and Spanish. Through the use of deep learning and pre-trained models, we aimed to enhance model performance. Our experimental results demonstrated competitive performance, with average macro-F1 scores of 63.02 and 58.39 for Subtasks 1 and 2, respectively. We found that the ERNIE 2.0 model achieved the highest scores for Subtask 1 and Subtask 2 in English, while the Multilingual DistilBERT model performed best for Subtask 2 in Spanish.

Official task results reveal a decline in macro-F1 scores for Subtask 1 compared to our experimental setup. This could be attributed to the presence of out-of-distribution samples caused by domain shift. The test set domains differed from those encountered during training, leading to limitations in capturing generalizable features due to the small training sample size.

Moreover, the challenging nature of Subtask 2, involving multi-class classification, resulted in most models only marginally surpassing chance-level performance, highlighting the complexity associated with this task.

Overall, our participation in the shared task contributed to the advancement of automated text identification techniques. It also sheds light on the performance of different models and data augmentation strategies.

8. Future Work

Future research can concentrate on model fine-tuning, addressing domain shift challenges, and utilising ensemble methods.

Due to the constraints imposed by our experiments and the timeline of the task, we were unable to explore model hyperparameter tuning [82, 83, 28]. We strongly believe that incorporating hyperparameter tuning would enhance the overall performance of the models and should be a key aspect to be explored in future tasks. Furthermore, we recommend the adoption of more advanced learning approaches in future tasks to enhance the reliability and robustness of the results obtained.

Deep neural networks have demonstrated remarkable success in learning from labelled data and achieving state-of-the-art performance in various Natural Language Processing tasks. However, learning from unlabelled data, particularly in the presence of domain shift, continues to pose challenges. To overcome this limitation, it is crucial to explore and employ unsupervised domain generalisation techniques [84, 85].

The task of Authorship Attribution, which involves multi-class classification, poses significant challenges, especially for transformer-based models when the ratio of authors to samples is

limited. Previous research has demonstrated that traditional methods, such as logistic regression combined with statistical feature extraction [86, 81], can outperform transformers in this context. A potential future direction is to investigate the ensemble of deep learning approaches with traditional methods to create a more robust and effective solution for the authorship attribution problem. By combining the strengths of both approaches, it may be possible to improve performance and overcome the limitations faced by transformers in this particular task.

Acknowledgments

This research is supported in part by the Defence Science and Technology Group, Australia and the Australian Research Council Discovery Project DP200101441.

References

[1] OpenAI, Introducing chatgpt, 2022. URL: https://openai.com/blog/chatgpt.

[2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[3] G. AI, Meet bard, 2023. URL: https://bard.google.com/.

[4] OpenAI, Gpt-4 technical report, 2023. arXiv:2303.08774.

[5] X. Yang, A. Chen, N. PourNejatian, H. C. Shin, K. E. Smith, C. Parisien, C. Compas, C. Martin, A. B. Costa, M. G. Flores, Y. Zhang, T. Magoc, C. A. Harle, G. Lipori, D. A. Mitchell, W. R. Hogan, E. A. Shenkman, J. Bian, Y. Wu, A large language model for electronic health records, npj Digital Medicine 5 (2022) 194. URL: https://doi.org/10.1038/s41746-022-00742-2. doi:10.1038/s41746-022-00742-2.

[6] H. U. Haq, V. Kocaman, D. Talby, Deeper clinical document understanding using relation extraction, arXiv preprint arXiv:2112.13259 (2021).

[7] H. U. Haq, V. Kocaman, D. Talby, Mining adverse drug reactions from unstructured mediums at scale, in: Multimodal AI in healthcare: A paradigm shift in health intelligence, Springer, 2022, pp. 361–375.

[8] D. Trautmann, A. Petrova, F. Schilder, Legal prompt engineering for multilingual legal judgement prediction, arXiv preprint arXiv:2212.02199 (2022).

[9] A. Blair-Stanek, N. Holzenberger, B. Van Durme, Can gpt-3 perform statutory reasoning?, arXiv preprint arXiv:2302.06100 (2023).

[10] L. Lumley, Large language models advance on financial services, 2023. URL: https://www.thebanker.com/Markets/Large-language-models-advance-on-financial-services.

[11] C. Haas, Introducing bloomberggpt, bloomberg's 50-billion parameter large language model, purpose-built from scratch for finance, 2023. URL: https://www.bloomberg.com/company/press/bloomberggpt-50-billion-parameter-llm-tuned-finance/.

[12] B. Delocski, Natural language processing and its applications in the finance sector, 2023. URL: https://www.edlitera.com/en/blog/posts/nlp-in-finance.

[13] E. Bagdasaryan, V. Shmatikov, Spinning language models: Risks of propaganda-as-a-service and countermeasures, in: 2022 IEEE Symposium on Security and Privacy (SP),

IEEE, 2022. URL: https://doi.org/10.1109%2Fsp46214.2022.9833572. doi:10.1109/sp46214.2022.9833572.
[14] E. Groll, Researchers: Large language models will revolutionize digital propaganda campaigns, 2023. URL: https://cyberscoop.com/large-language-models-influence-operatio/.
[15] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, Y. Choi, Defending against neural fake news, Advances in neural information processing systems 32 (2019).
[16] C. Goldschmidt, Council post: Ai-generated reviews threaten business reputations, 2019. URL: https://www.forbes.com/sites/forbestechcouncil/2019/04/04/ai-generated-reviews-threaten-business-reputations/?sh=2f7bab347eb1.
[17] M. Stella, E. Ferrara, M. De Domenico, Bots increase exposure to negative and inflammatory content in online social systems, Proceedings of the National Academy of Sciences 115 (2018) 12435–12440.
[18] A. Bessi, E. Ferrara, Social bots distort the 2016 us presidential election online discussion, First monday 21 (2016).
[19] D. R. Cotton, P. A. Cotton, J. R. Shipway, Chatting and cheating: Ensuring academic integrity in the era of chatgpt, Innovations in Education and Teaching International (2023) 1–12.
[20] J. P. Wahle, T. Ruas, F. Kirstein, B. Gipp, How large language models are transforming machine-paraphrased plagiarism, arXiv preprint arXiv:2210.03568 (2022).
[21] F. R. Elali, L. N. Rachid, Ai-generated research paper fabrication and plagiarism in the scientific community, Patterns 4 (2023).
[22] K. C. McLean, M. A. Fournier, The content and processes of autobiographical reasoning in narrative identity, Journal of research in personality 42 (2008) 527–545.
[23] E. Derner, K. Batistič, Beyond the safeguards: Exploring the security risks of chatgpt, arXiv preprint arXiv:2305.08005 (2023).
[24] M. Gault, Ai spam is already flooding the internet and it has an obvious tell, 2023. URL: https://www.vice.com/en/article/5d9bvn/ai-spam-is-already-flooding-the-internet-and-it-has-an-obvious-tell.
[25] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
[26] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).
[27] A. Uchendu, T. Le, K. Shu, D. Lee, Authorship attribution for neural text generation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 8384–8395.
[28] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: Chinese Computational Linguistics: 18th China National Conference, CCL 2019, Kunming, China, October 18–20, 2019, Proceedings 18, Springer, 2019, pp. 194–206.
[29] S. González-Carvajal, E. C. Garrido-Merchán, Comparing BERT against traditional machine learning text classification, CoRR abs/2005.13012 (2020). URL: https://arxiv.org/abs/2005.13012. arXiv:2005.13012.
[30] T. Fagni, F. Falchi, M. Gambini, A. Martella, M. Tesconi, Tweepfake: About detecting deepfake tweets, Plos one 16 (2021) e0251415.

[31] Turnitin, Ai writing: Ai tools, 2023. URL: https://www.turnitin.com/solutions/ai-writing.
[32] E. Tian, Home, 2023. URL: https://gptzero.me/.
[33] OpenAI, Gpt-4, chatgpt amp; ai detector by zerogpt: Detect openai text, 2023. URL: https://www.zerogpt.com/.
[34] OpenAI, New ai classifier for indicating ai-written text, 2023. URL: https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text.
[35] NeuralText, Ai text detector app, 2023. URL: https://gptradar.com/.
[36] K. Jones, J. R. Nurse, S. Li, Are you robert or roberta? deceiving online authorship attribution models using neural text generators, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 16, 2022, pp. 429–440.
[37] A. Uchendu, T. Le, D. Lee, Attribution and obfuscation of neural text authorship: A data mining perspective, arXiv preprint arXiv:2210.10488 (2022).
[38] C. A. Gao, F. M. Howard, N. S. Markov, E. C. Dyer, S. Ramesh, Y. Luo, A. T. Pearson, Comparing scientific abstracts generated by chatgpt to original abstracts using an artificial intelligence output detector, plagiarism detector, and blinded human reviewers, bioRxiv (2022). URL: https://www.biorxiv.org/content/early/2022/12/27/2022.12.23.521610. doi:10.1101/2022.12.23.521610.
[39] R. Gagiano, M. M.-H. Kim, X. Zhang, J. Biggs, Robustness analysis of grover for machine-generated news detection, in: Proceedings of the The 19th Annual Workshop of the Australasian Language Technology Association, Australasian Language Technology Association, Online, 2021, pp. 119–127. URL: https://aclanthology.org/2021.alta-1.12.
[40] A. Uchendu, Z. Ma, T. Le, R. Zhang, D. Lee, Turingbench: A benchmark environment for turing test in the age of neural text generation, arXiv preprint arXiv:2109.13296 (2021).
[41] S. M. Jiménez-Zafra, F. Rangel, M. Montes-y Gómez, Overview of IberLEF 2023: Natural Language Processing Challenges for Spanish and other Iberian Languages, Procesamiento del Lenguaje Natural 71 (2023).
[42] A. M. Sarvazyan, J. Á. González, M. Franco Salvador, F. Rangel, B. Chulvi, P. Rosso, Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains, in: Procesamiento del Lenguaje Natural, Jaén, Spain, 2023.
[43] A. McCallum, K. Nigam, et al., A comparison of event models for naive bayes text classification, in: AAAI-98 workshop on learning for text categorization, volume 752, Madison, WI, 1998, pp. 41–48.
[44] S. Dreiseitl, L. Ohno-Machado, Logistic regression and artificial neural network classification models: a methodology review, Journal of biomedical informatics 35 (2002) 352–359.
[45] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, Journal of machine learning research 2 (2001) 45–66.
[46] B. Xu, X. Guo, Y. Ye, J. Cheng, An improved random forest classifier for text categorization., J. Comput. 7 (2012) 2913–2920.
[47] S. Jiang, G. Pang, M. Wu, L. Kuang, An improved k-nearest-neighbor algorithm for text categorization, Expert Systems with Applications 39 (2012) 1503–1509.
[48] K. Shah, H. Patel, D. Sanghvi, M. Shah, A comparative analysis of logistic regression, random forest and knn models for the text classification, Augmented Human Research 5 (2020) 1–16.

[49] T. Pranckevičius, V. Marcinkevičius, Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification, Baltic Journal of Modern Computing 5 (2017) 221.
[50] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.
[51] Y. Freund, R. Schapire, N. Abe, A short introduction to boosting, Journal-Japanese Society For Artificial Intelligence 14 (1999) 1612.
[52] R. A. Stein, P. A. Jaques, J. F. Valiati, An analysis of hierarchical text classification using word embeddings, Information Sciences 471 (2019) 216–232.
[53] Z. Qi, The text classification of theft crime based on tf-idf and xgboost model, in: 2020 IEEE International conference on artificial intelligence and computer applications (ICAICA), IEEE, 2020, pp. 1241–1246.
[54] C. Tang, N. Luktarhan, Y. Zhao, An efficient intrusion detection method based on lightgbm and autoencoder, Symmetry 12 (2020) 1458.
[55] E.-A. Minastireanu, G. Mesnita, Light gbm machine learning algorithm to online click fraud detection, J. Inform. Assur. Cybersecur 2019 (2019) 263928.
[56] S. Bloehdorn, A. Hotho, Boosting for text classification with semantic features, in: Advances in Web Mining and Web Usage Analysis: 6th International Workshop on Knowledge Discovery on the Web, WebKDD 2004, Seattle, WA, USA, August 22-25, 2004, Revised Selected Papers 6, Springer, 2006, pp. 149–166.
[57] J. Wu, Introduction to convolutional neural networks, National Key Lab for Novel Software Technology. Nanjing University. China 5 (2017) 495.
[58] L. Medsker, L. C. Jain, Recurrent neural networks: design and applications, CRC press, 1999.
[59] D. Yogatama, C. Dyer, W. Ling, P. Blunsom, Generative and discriminative text classification with recurrent neural networks, arXiv preprint arXiv:1703.01898 (2017).
[60] P. Bharadwaj, Z. Shao, Fake news detection with semantic features and text mining, International Journal on Natural Language Computing (IJNLC) Vol 8 (2019).
[61] Y. Zhou, B. Xu, J. Xu, L. Yang, C. Li, Compositional recurrent neural networks for chinese short text classification, in: 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI), IEEE, 2016, pp. 137–144.
[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).
[63] J. R. Medina, J. Kalita, Parallel attention mechanisms in neural machine translation, in: 2018 17th IEEE international conference on machine learning and applications (ICMLA), IEEE, 2018, pp. 547–552.
[64] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.
[65] S. González-Carvajal, E. C. Garrido-Merchán, Comparing bert against traditional machine learning text classification, arXiv preprint arXiv:2005.13012 (2020).
[66] J. Briskilal, C. Subalalitha, An ensemble model for classifying idioms and literal texts using bert and roberta, Information Processing & Management 59 (2022) 102756.

[67] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning–based text classification: a comprehensive review, ACM computing surveys (CSUR) 54 (2021) 1–40.
[68] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).
[69] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, H. Wang, Ernie 2.0: A continual pre-training framework for language understanding, in: Proceedings of the AAAI conference on artificial intelligence, volume 34, 2020, pp. 8968–8975.
[70] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, Advances in neural information processing systems 32 (2019).
[71] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, et al., Bloom: A 176b-parameter open-access multilingual language model, arXiv preprint arXiv:2211.05100 (2022).
[72] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).
[73] H. Face, Hugging face – transformer package, 2023. URL: https://huggingface.co/docs/transformers/v4.29.1/en/main_classes/trainer#transformers.TrainingArguments.
[74] H. Face, Hugging face – the ai community building the future., 2023. URL: https://huggingface.co/.
[75] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text classification algorithms: A survey, Information 10 (2019) 150.
[76] G. Song, Y. Ye, X. Du, X. Huang, S. Bie, Short text classification: a survey., Journal of multimedia 9 (2014).
[77] I. Alsmadi, K. H. Gan, Review of short-text classification, International Journal of Web Information Systems 15 (2019) 155–182.
[78] H. Face, Helsinki-nlp - opus-mt-en-es, 2023. URL: https://huggingface.co/Helsinki-NLP/opus-mt-en-es.
[79] H. Face, Salesken - opus-mt-es-en, 2023. URL: https://huggingface.co/salesken/translation-spanish-and-portuguese-to-english.
[80] M. Mosbach, M. Andriushchenko, D. Klakow, On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines, arXiv preprint arXiv:2006.04884 (2020).
[81] J. Tyo, B. Dhingra, Z. C. Lipton, On the state of the art in authorship attribution and authorship verification, arXiv preprint arXiv:2209.06869 (2022).
[82] J. Howard, S. Ruder, Universal language model fine-tuning for text classification, arXiv preprint arXiv:1801.06146 (2018).
[83] N. Kant, R. Puri, N. Yakovenko, B. Catanzaro, Practical text classification with large pre-trained language models, arXiv preprint arXiv:1812.01207 (2018).
[84] A. Ramponi, B. Plank, Neural unsupervised domain adaptation in nlp—a survey, arXiv preprint arXiv:2006.00632 (2020).
[85] X. Liu, C. Yoo, F. Xing, H. Oh, G. El Fakhri, J.-W. Kang, J. Woo, et al., Deep unsupervised domain adaptation: A review of recent advances and perspectives, APSIPA Transactions on Signal and Information Processing 11 (2022).
[86] M. Altakrori, J. C. K. Cheung, B. C. M. Fung, The topic confusion task: A novel evaluation

scenario for authorship attribution, in: Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 4242–4256. URL: https://aclanthology.org/2021.findings-emnlp.359. doi:10.18653/v1/2021.findings-emnlp.359.