future internet

MDPI

Article

Artificial Intelligence vs. Human: Decoding Text Authenticity with Transformers

Daniela Gifu [1,2,*] and Covaci Silviu-Vasile [2]

[1] Institute of Computer Science, Romanian Academy—Iași Branch, Codrescu 2, 700481 Iași, Romania
[2] George Emil Palade University of Medicine, Pharmacy, Science, and Technology of Târgu Mureș, Gheorghe Marinescu 38, 540142 Târgu Mureș, Romania; covaci.silviu-vasile.23@stud.umfst.ro
* Correspondence: daniela.gifu@iit.academiaromana-is.ro; Tel.: +40-742050673

Abstract: This paper presents a comprehensive study on detecting AI-generated text using transformer models. Our research extends the existing RODICA dataset to create the Enhanced RODICA for Human-Authored and AI-Generated Text (ERH) dataset. We enriched RODICA by incorporating machine-generated texts from various large language models (LLMs), ensuring a diverse and representative corpus. Methodologically, we fine-tuned several transformer architectures, including BERT, RoBERTa, and DistilBERT, on this dataset to distinguish between human-written and AI-generated text. Our experiments examined both monolingual and multilingual settings, evaluating the model's performance across diverse datasets such as M4, AICrowd, Indonesian Hoax News Detection, TURNBACKHOAX, and ERH. The results demonstrate that RoBERTa-large achieved superior accuracy and F-scores of around 83%, particularly in monolingual contexts, while DistilBERT-multilingual-cased excelled in multilingual scenarios, achieving accuracy and F-scores of around 72%. This study contributes a refined dataset and provides insights into model performance, highlighting the transformative potential of transformer models in detecting AI-generated content.

Keywords: large language models; natural language processing; content creation; text authenticity

Academic Editors: Gianluigi Ferrari and Paulo Ferreira

Received: 23 July 2024
Revised: 6 November 2024
Accepted: 13 January 2025
Published: 16 January 2025

Citation: Gifu, D.; Silviu-Vasile, C. Artificial Intelligence vs. Human: Decoding Text Authenticity with Transformers. *Future Internet* **2025**, *17*, 38. https://doi.org/10.3390/fi17010038

1. Introduction

The proliferation of large language models (LLMs), particularly those developed by OpenAI, has significantly blurred the lines between human and machine-generated content, amplifying concerns about text authenticity [1–3]. In today's landscape, where misinformation spreads across various domains [4–6], distinguishing between human-written and machine-generated text is critical to combating the risks associated with deceptive content [7,8]. While prior research has often focused on detecting text generated by specific LLMs or domain-specific models (e.g., ChatGPT), our study addresses the broader challenge of distinguishing between human and machine-generated text [9].

Transformer models, such as Bidirectional Encoder Representations from Transformers (BERT) [10], have emerged as powerful tools in Natural Language Processing (NLP) [11,12], demonstrating impressive capabilities across a variety of tasks, including text generation (TG) [12]. The Generative Pretrained Transformer 3 (GPT-3), in particular, has excelled across numerous NLP tasks due to its self-attention mechanism, which allows it to weigh each word in the context of a sentence differently, capturing intricate relationships between words and their meanings. As transformer models continue to advance, they have become

foundational for large-scale self-supervised learning systems [13,14], revolutionizing other AI and machine learning fields, including time series analysis [15].

However, the increasing fluency of these models raises challenges in effectively discerning between human and machine-generated text [16]. Our research centers on using transformer models combined with Bidirectional Long Short-Term Memory (BiLSTM) to predict the source of text and address the challenge of text authenticity detection. This work not only advances text authenticity detection systems but also highlights the versatility and effectiveness of transformer-based methodologies in NLP applications.

The main research question of this paper is as follows: How efficient are transformers in building a classifier that can accurately detect human-written text from machine-generated text? Through experimental analysis and methodology evaluation, we aim to provide valuable insights into this question.

The structure of this paper is as follows: Section 2 positions this study in the context of existing literature. Section 3 outlines the dataset and methodology based on transformers. Section 4 presents the usability and efficiency of the proposed method through a series of tests, followed by concluding remarks in the final section.

Current Survey Mission

This paper surveys existing techniques for classifying texts as human-written or machine-generated, identifies their limitations, and proposes models leveraging contextual understanding to improve the accuracy and reliability of classification systems.

The main contributions of our research are as follows:

- **Refinement and Extension of the ERH Dataset**: Originally published as RODICA in 2016 [17,18], the ERH dataset has been significantly expanded for this study. It now includes additional machine-generated texts from multiple large language models (LLMs) in English, Romanian, and Hungarian. This expanded dataset plays a crucial role in our research, offering a comprehensive resource for detecting AI-generated text across diverse languages.
- **Implementation of Classification Models**: We implemented state-of-the-art transformer-based classification models, such as BERT-base, RoBERTa-base, RoBERTa-large, DistilBERT-base-uncased, XLM-RoBERTa-base [19], BERT-base-multilingual-cased, and DistilBERT-base-multilingual-cased, along with classic machine learning models. Our models are specifically tailored to automatically classify human versus AI-generated texts in multiple languages (English, Romanian, and Hungarian), offering a robust and novel multilingual solution.

In addition, we release the extended dataset and codebase as open-source resources for the community to support further research in this domain (available at Papers with Code (https://paperswithcode.com/datasets), AI Crowd (https://www.aicrowd.com/challenges/kiit-ai-mini-blitz/problems/fake-news-detection), Mendeley Data (https://data.mendeley.com/datasets/hps7rcbwm6/1), GitHub (https://github.com/jibranfawaid/turnbackhoax-dataset/tree/main?tab=readme-ov-file#turnbackhoax-dataset), RELATE (racai.ro) (https://relate.racai.ro/index.php?path=repository/resource&resource=rodica), and ERH, (https://github.com/SilviuCovaci/AIContentDetector/tree/main/ERH), accessed on 22 July 2024), along with the codebase (available at GitHub (https://github.com/SilviuCovaci/AIContentDetector), accessed on 22 July 2024).

2. Background

The rapid advancement of AI technologies has significantly enhanced the capabilities of text generation models, blurring the distinction between human and machine-generated

content. This evolution marks a shift from traditional machine learning techniques to more sophisticated models that excel in handling complex language tasks.

In the past, text processing and classification tasks relied on various machine learning techniques such as Convolutional Neural Networks (CNNs) [19], Long Short-Term Memory (LSTM) networks [20], and hybrid models like CNN-LSTM [21–24]. While these models, along with Support Vector Machines (SVMs) [25] and Decision Trees (DTs) [26], provided notable performance improvements, they often struggled with capturing intricate contextual relationships and generating coherent text.

However, recent years have seen a transformative shift with the advent of transformer-based models in Natural Language Processing (NLP). Models like BERT [10], RoBERTa [27], and DistilBERT [28] have set new benchmarks across various NLP tasks, including Machine Translation (MT) [13,29], Question Answering (QA) [30,31], Text Summarization (TS) [32,33], and Text Classification (TC) [34,35]. These advancements have significantly improved the ability to discern between human-written and machine-generated text [36,37].

Transformers excel in capturing complex contextual relationships and generating coherent text, surpassing the capabilities of traditional models like CNNs and Bi-LSTMs [38]. This capability is particularly crucial for detecting AI-generated content, where subtle differences between human and machine-generated text pose significant challenges.

Recent studies highlight the effectiveness of transformer models in text detection tasks, highlighting their superiority over traditional methods by leveraging their deep understanding and representation of intricate language patterns [39]. Moreover, fine-tuned transformer models have demonstrated substantial improvements in detection accuracy compared to earlier machine learning techniques [40].

In the subsequent sections, we provide a detailed description of the methods and materials used in our study, including the specific transformer models employed and their application in detecting AI-generated text.

2.1. BERT

Bidirectional Encoder Representations from Transformers (BERT), introduced by Google AI in 2018 [10], represents a groundbreaking development in Natural Language Processing (NLP). Unlike unidirectional models, BERT captures context bidirectionally in a text sequence, enhancing its performance across a broad spectrum of NLP tasks.

BERT's architecture extends the original Transformer model, incorporating multiple layers, extensive feed-forward networks, and numerous attention heads. Pre-trained using Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) tasks (Figure 1), BERT's input embeddings comprise token embeddings, segmentation embeddings, and position embeddings.

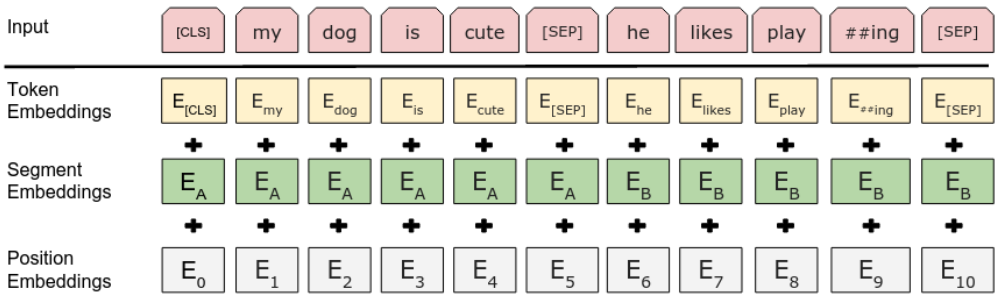

Figure 1. BERT input representation.

The MLM task involves predicting the original vocabulary ID of masked words based on contextual clues, with random masking applied during training. The NSP task trains

BERT to discern relationships between sentence pairs, predicting whether they follow consecutively within a document.

2.2. RoBERTa

RoBERTa [32], an optimized variant of BERT, incorporates several enhancements, including dynamic masking and training on extensive datasets. These improvements significantly bolster RoBERTa's ability to generate and comprehend contextually rich text. For instance, RoBERTa employs a larger batch size of 2000 and undergoes 500,000 training steps, optimizing adaptation to dynamic masking compared to BERT's 256 batch size and 1 million training steps [35,40].

RoBERTa utilizes Byte Pair Encoding (BPE) for tokenization, a method that breaks down text into subwords using advanced statistical techniques based on bytes rather than Unicode characters, thus enhancing encoding efficiency [41,42]. Its extended training on diverse datasets such as Common Crawl and WebText contributes to superior performance across a wide array of NLP applications [27,43].

2.3. DistilBERT

DistilBERT, developed by Hugging Face, offers a streamlined version of BERT with maintained effectiveness, faster inference times, and reduced computational demands [28,44]. Achieving this efficiency by reducing transformer layers and omitting certain components, DistilBERT strikes a balance between performance and resource efficiency, making it suitable for scenarios with limited computational resources while still delivering high-quality results.

3. Materials and Methods

In this section, we delve into the methodologies and tools employed to classify either human-written or machine-generated texts. Our focus is on leveraging a range of AI text classifiers developed to tackle this specific challenge.

We begin by outlining the existing corpus used for classification and then provide an overview of the key AI text classifiers that have been developed to address the task of differentiating between human and machine-generated content.

We explore how these classifiers have been designed and fine-tuned to improve their performance in distinguishing between texts produced by humans and those generated by advanced AI systems. By examining these methods, we aim to provide insights into their effectiveness and the criteria that contribute to successful text classification. This section will also compare the performance of these classifiers, offering a comprehensive view of their capabilities and the practical implications of their use in real-world scenarios.

3.1. Dataset

The corpus for this study consists of multiple datasets with comparable text lengths, including both machine-generated and human-written content. Experiments were conducted iteratively across all datasets to provide a comprehensive overview. To ensure that the detector generalizes well across various domains and writing styles, the human dataset includes texts from the following diverse domains:

- **M4 dataset** (https://paperswithcode.com/datasets, accessed on 22 July 2024): This dataset contains human-written text from sources such as Wikipedia, WikiHow [45], Reddit (ELI5), arXiv, and PeerRead [46] for English, as well as news articles for Urdu, RuATD [47] for Russian, and Indonesian news articles. Machine-generated text in this dataset is sourced from several multilingual Large Language Models (LLMs), including:

- **ChatGPT**: A widely recognized conversational AI model developed by OpenAI, based on the GPT-4 architecture.
- **Text-Davinci-003**: An earlier model from OpenAI's GPT-3 family, known for its text generation capabilities.
- **LLaMa [48]** (https://github.com/meta-llama/llama, accessed on 22 July 2024): A smaller and more efficient model developed by Meta, fine-tuned for multilingual tasks.
- **Flan-T5 [49]**: A fine-tuned version of Google's T5 model, optimized for natural language understanding and generation tasks.
- **Cohere**: A text generation model from Cohere AI, fine-tuned for various NLP tasks.
- **Dolly-v2**: An open-source large language model developed by Databricks, trained for text generation tasks.
- **BLOOMZ [50]**: A multilingual language model fine-tuned for zero-shot tasks, part of the BLOOM family, designed for generating and understanding text in multiple languages.

- **AI Crowd FakeNews Dataset** (https://www.aicrowd.com/challenges/kiit-ai-mini-blitz/problems/fake-news-detection, accessed on 22 July 2024): This dataset contains texts from various news articles and texts generated by OpenAI's GPT-2. The dataset was published by AI Crowd as part of the KIIT AI (mini)Blitz Challenge.
- **Indonesian Hoax News Detection Dataset** (INDONESIAN HOAX NEWS DETECTION DATASET—Mendeley Data, accessed on 22 July 2024) [51]: This dataset contains valid and hoax news articles in Indonesian. It is structured in CSV format, with two columns: text and label.
- **TURNBACKHOAX Dataset** (https://github.com/jibranfawaid/turnbackhoax-dataset/tree/main?tab=readme-ov-file#turnbackhoax-dataset, accessed on 22 July 2024): This dataset includes valid and hoax news articles in Indonesian. It is structured in CSV format, with three columns: label, headline, and body.
- **ERH** (https://relate.racai.ro/index.php?path=repository/resource&resource=rodica and GitHub https://github.com/SilviuCovaci/AIContentDetector/commit/2839b50fe35f44e407dd7744828a657914cf4d4f, accessed on 22 July 2024): This dataset comprises news articles in English, Romanian, and Hungarian. It is structured in JSON format with the extension JSON Lines (JSONL).

Tables 1 and 2 present the dataset collections.

Table 2 provides an overview of the dataset statistics used for training and testing in our study, highlighting the differences in dataset sizes and their contribution to our classification tasks. Note that the information for the M4 dataset, including the table format, has been adapted from Wang et al. (2023) [52] to fit the context of our study. Additionally, the symbol # in the table header refers to the number of records used for training and testing. Specifically: # Training Records: Indicates the total number of data samples used to train the model. # Testing Records: Indicates the total number of data samples used to evaluate the model's performance.

Table 1. Data Sources in M4 Dataset.

Source	Lang. [1]	Only Human	Source-Generated Data						
			Human	Davinci003	ChatGPT	Cohere	Dolly-v2	BLOOM	Total
Wikipedia	EN	6,458,670	3.000	3.000	2.995	2.336	2.702	3.000	17,033
Reddit ELIS	EN	558,669	3.000	3.000	3.000	3.000	3.000	3.000	18,000
WikiHow	EN	31,102	3.000	3.000	3.000	3.000	3.000	3.000	18,000
PeerRead	EN	5.798	5.798	2.344	2.344	2.344	2.344	2.344	17,518
arXiv abstract	EN	2,219,423	3.000	3.000	3.000	3.000	3.000	3.000	18,000
Baike/Web OA	ZH	113,313	3.000	3.000	3.000	-	-	-	9.000
RuATD	RU	75,291	3.000	3.000	3.000	-	-	-	9.000
Urdu-news	UR	107,881	3.000	-	3.000	-	-	-	9.000
id_newspapers_2018	ID	499,164	3.000	-	3.000	-	-	-	6.000
Arabic-Wikipedia	AR	1,209,042	3.000	-	3.000	-	-	-	6.000
True and Fake News	BG	94,000	3.000	3.000	3.000	-	-	-	9.000
Total			35,798	23,344	32,339	13,680	14,046	14,344	133,551

[1] Here are the abbreviations provided for the ISO 639-1 language codes: English, EN; Chinese, ZH; Russian, RU; Urdu, UR; Indonesian, ID; Arabic, AR; Bulgarian, BG.

Table 2. Dataset statistics.

Language Approach	# Training Records	# Testing Records
M4—Monolingual	119.757	5.000
AICrowd—Monolingual	232.003	38.666
M4—Multilingual	172.417	4.000
Indonesian Hoax News Detection—Multilingual	600	250
TURNBACKHOAX Dataset—Multilingual	800	316
ERH Dataset—Multilingual	316	3.000

The difference in the number of training and testing records between the M4 and AI Crowd datasets, particularly in the monolingual approach, reflects the inherent characteristics and design of these datasets.

Reasoning:

- **Dataset Availability and Scope**: The AICrowd dataset was chosen because of its larger and more diverse set of examples, which is crucial for training robust models. Its comprehensive nature provides a broad representation of different text styles and topics, helping improve the model's generalization across varied input data. The M4 dataset, while valuable, has a more focused scope, leading to fewer examples, especially for testing.
- **Balancing Model Evaluation**: The larger volume of test records from AICrowd allows for a more thorough assessment of the model's performance across diverse examples. This is critical for evaluating generalization, especially when the training process uses a broad dataset. Meanwhile, the smaller test set from M4 is curated for quality, allowing for a more targeted evaluation of the model's capabilities.
- **Strategic Test Set Size**: The M4 test set focuses on quality over quantity, ensuring targeted and insightful evaluation. The larger AICrowd test set evaluates robustness across varied data.

In summary, the AICrowd test set is used to evaluate the model's robustness, while the M4 test set targets quality, ensuring comprehensive and nuanced evaluation. The ERH corpus introduces additional challenges, including new languages, domains, and generators, simulating real-world scenarios. Notably, BLOOMZ outputs are used in monolingual language tests, enhancing the model's readiness for real-world applications.

The input data are organized as JavaScript Object Notation (JSON) records in files with the extension JSON Lines (JSONL).

The structure of each record is very straightforward and intuitive.

```
{
  id -> identifier of the example,
  label -> label (human text:  0, machine text:  1,),
  text -> text generated by a machine or written by a human,
  model -> model that generated the data,
  source -> source (Wikipedia, Wikihow, Peerread, Reddit, Arxiv) on English
or language (Arabic, Russian, Chinese, Indonesian, Urdu, Bulgarian, German,
Bulgarian, Romanian, Hungarian)
}
```

Figures 2 and 3 illustrate the structure of the datasets used for training and testing in our study, offering insights into the composition and organization of the data.

text	label	model	source	id
Forza Motorsport is a popular racing game that...	1	chatGPT	wikihow	0
Buying Virtual Console games for your Nintendo...	1	chatGPT	wikihow	1
Windows NT 4.0 was a popular operating system ...	1	chatGPT	wikihow	2
How to Make Perfume\n\nPerfume is a great way ...	1	chatGPT	wikihow	3
How to Convert Song Lyrics to a Song"\n\nConve...	1	chatGPT	wikihow	4
...	...	...	...	...
During the Cold War, the United States was po...	1	cohere	reddit	172412
The "continuity thesis" is the idea that ther...	1	cohere	reddit	172413
In the early Middle Ages, the pagan Norse wer...	1	cohere	reddit	172414
There are many similarities between the langu...	1	cohere	reddit	172415
News of Christopher Columbus' voyage to the N...	1	cohere	reddit	172416

(**a**) The structure and composition of the monolingual training dataset.

text	label	model	source	id
This lightening spray calls for dried chamomi...	0	human	wikihow	69365
We consider the problem of mass transport cl...	0	human	arxiv	113114
\nThe paper "Learning to Skim Text" presents a...	1	cohere	peerread	47161
Tesla's solar shingles and Powerwall are innov...	1	chatGPT	reddit	38209
Have you accidentally dropped your precious iP...	1	chatGPT	wikihow	1368
...	...	...	...	...
Josephine "Joyce" Luther Kennard (born May 6, ...	0	human	wikipedia	76820
The fractional Brownian motion with index $\...	0	human	arxiv	110268
Strictly subadditive, subadditive and weakly...	0	human	arxiv	103694
How to Get Into UPenn: Excelling Academically ...	1	chatGPT	wikihow	860
Well, from what little information I have gath...	1	chatGPT	reddit	15795

(**b**) The structure and composition of the multilingual training dataset.

Figure 2. Mono- and multilingual Training Dataset.

text	label	model	source	id
Giving gifts should always be enjoyable. Howe...	1	bloomz	wikihow	0
Yveltal (Japanese: ユベルタル) is one of the main a...	1	bloomz	wikihow	1
If you'd rather not annoy others by being rude...	1	bloomz	wikihow	2
If you're interested in visiting gravesite(s) ...	1	bloomz	wikihow	3
The following are some tips for becoming succe...	1	bloomz	wikihow	4
...	...	...	...	...
The paper deals with an interesting applicatio...	0	human	peerread	4995
This manuscript tries to tackle neural network...	0	human	peerread	4996
The paper introduced a regularization scheme t...	0	human	peerread	4997
Inspired by the analysis on the effect of the ...	0	human	peerread	4998
\n- You definitely need to report misclassific...	0	human	peerread	4999

Figure 3. Testing Dataset.

Figure 2 shows how the training datasets are organized for both monolingual and multilingual scenarios. It details the distribution of text samples, including the breakdown by language and type (human-written vs. machine-generated). The figure provides insights into the dataset's design, highlighting key features such as sample size, language coverage, and the balance between different classes.

Figure 3 presents the structure of the testing datasets used to evaluate the performance of the classifiers. It outlines the composition of the test sets, similar to the training datasets, but focuses on the evaluation phase. The figure includes information on the distribution of test samples, the language diversity, and the proportion of human-written versus machine-generated texts.

There are mainly three major differences if we compare the datasets used for training the models and the dataset that will be used for final evaluation:

- The task formulation is different;

- Human text was upsampled to balance the data;
- New and surprising domains, generators, and languages will appear in the test sets. Real test sets will not include information about generators, domains, and languages.

Nevertheless, the test dataset includes BLOOMZ (BLOOMZ, a variant of BLOOM model, supports 46 human languages. Hugging Face reports that the 7 billion-parameter BLOOMZ runs three times faster on the Intel Habana Gaudi2 compared to the A100-80G) outputs (for monolingual language) that are not included in the training set. Moreover, the model is prepared for real-world application scenarios.

3.2. System Overview

The system's architecture, shown in Figure 4, is based on BERT-based transformers, including BERT, RoBERTa, and DistilBERT, using the HuggingFace library. These models were pretrained on extensive generic datasets [10,45,47,53,54] and fine-tuned for specific NLP tasks, including text classification, named entity recognition, and sentiment analysis [48,55].

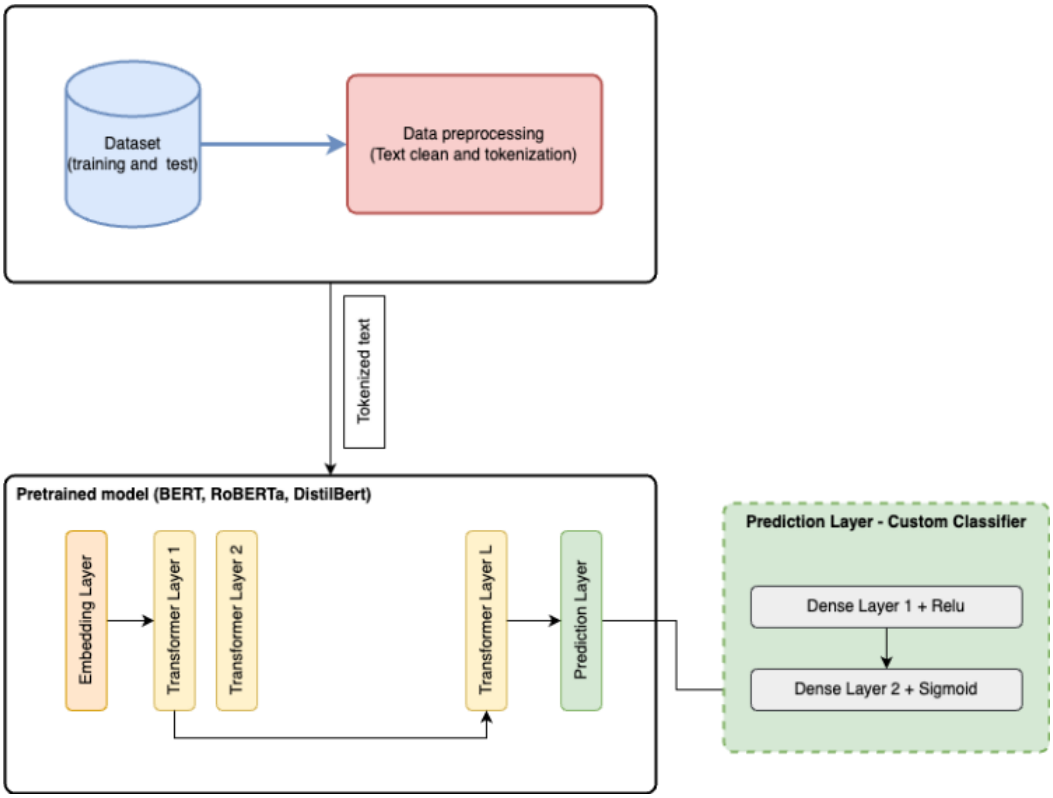

Figure 4. Architecture.

The architecture combines Hugging Face's Transformers library with PyTorch and Scikit-Learn [56] for implementation. To enhance the pretrained models, we implemented a custom classifier with two dense layers:

- **First Dense Layer**: Matches the output dimensions of the pretrained models—768 neurons for "base" versions and 1024 neurons for "large" versions.
- **Second Dense Layer**: Contains 32 neurons for "base" versions and 8 neurons for "large" versions.
- **Output Layer**: A single neuron with a sigmoid activation function to produce probabilities between 0 and 1.

In our neural network:

- **Hidden Layers**: Used the Rectified Linear Unit (ReLU) activation function.
- **Output Layer**: Used the sigmoid activation function.
- **Optimizer**: AdamW with a learning rate of 1×10^{-5}.

For the monolingual experiments, we evaluated the following models:

- **Monolingual Setups**: BERT-base, RoBERTa-base, RoBERTa-large, and Distil-BERT-base-uncased.
- **Multilingual Setups**: XLM-RoBERTa-base, BERT-base-multilingual-cased, and DistilBERT-base-multilingual-cased.

To expedite training, GPUs were utilized for both model training and inference. All experiments were conducted on a Mac Studio machine.

4. Results

Our experimental framework involved preprocessing, feature engineering, and modeling using various transformer architectures.

- **Preprocessing**
  We developed a custom PyTorch DataSet class to handle data loading and perform essential preprocessing tasks:
  (1) Text Cleanup: This step involved removing HTML tags, special characters (e.g., #, @), punctuation, and multiple spaces to ensure the text was clean and consistent.
  (2) Basic preprocessing: Tokenization. Initially, we implemented a basic tokenization step as part of the preprocessing. However, after further consideration, we recognized that all the transformer models used in our experiments come with their own specialized tokenizers, which are optimized for their respective architectures. Consequently, we opted to rely on the model-specific tokenizers provided by the Hugging Face Transformers library during the modeling stage. This approach ensures that the text is tokenized in a way that is consistent with the model's expectations, avoiding any potential negative impact on the representations generated for the task.
- **Feature Engineering**
  We integrated Bag of Words (BoW) and Word2Vec models into our feature engineering process. BoW provided sparse representations by counting word occurrences, which captured frequency-based features of the text. Word2Vec offered dense vector representations that encapsulate semantic relationships between words. These features were used alongside the outputs of the transformer models, adding an additional layer of information that could enhance the model's understanding of both the syntactic and semantic aspects of the text. The inclusion of these features aimed to improve the model's overall performance by providing a more nuanced representation of the text.
- **Modeling**
  We experimented with several pretrained transformer models, including BERT-base, RoBERTa-base, RoBERTa-large, DistilBERT-base-uncased, XLM-RoBERTa-base, BERT-base-multilingual-cased, and DistilBERT-base-multilingual-cased. Each model's built-in tokenizer was used to ensure consistency and optimize the text representation according to the specific architecture. The models were combined with a custom classifier consisting of three dense layers, each with a varying number of neurons, to refine the predictions based on the features extracted from both the transformers and the feature-engineering step.
  As a baseline, we utilized the RoBERTa-base pretrained model, which was fine-tuned with a sequence classification head. This model was trained and evaluated using the same dataset mentioned earlier. Table 3 summarizes the hyperparameters used for the baseline model. We employed Cross-Entropy loss as the loss function due

to its effectiveness in binary classification tasks. The model's final output layer uses a Sigmoid function to produce probabilities between 0 (no AI-generated text) and 1 (AI-generated text). The Adaptive Moment Estimation (AdamW) optimizer [57], an improved version of Adaptive Moment Estimation (Adam), was chosen to fine-tune the learning rates across different parameters, ensuring efficient and effective convergence during training. Here, with a learning rate set to 2×10^{-5}.

Table 3. Baseline model: Hyperparameter Optimization.

Hyperparameter	Values
Learning rate	2×10^{-5}
Batch Size	16
Epochs	3
Weight decay	0.01

Table 4 outlines the hyperparameter optimization for the fine-tuned model.

Table 4. Fine-tuned model: Hyperparameter Optimization.

Hyperparameter	Values
Learning rate	1×10^{-5}
Batch Size	8
Epochs	5
Optimezer	AdamW
Activation Function (hidden layers)	ReLU
Activation Function (output layer)	Sigmoid

- **Prediction**
  The model outputs probabilities between 0 and 1, with a 50% threshold used for binary classification. Predictions were generated on a test dataset, which included test IDs and sample targets. The results were stored in a prediction file for further analysis. For evaluation, we used sklearn.metrics to calculate Accuracy (Acc), Precision (P), Recall (R), and F-score (also known as F1 score). This provided a comprehensive assessment of the model's performance across all relevant metrics.

For the multilingual subtask, we employed different pretrained models optimized for multilingual tasks, recognizing the specific challenges associated with handling multiple languages.

The experiments were conducted on a Mac Studio machine equipped with Apple's M1 Max Chip, which features a 10-core CPU, a 24-core GPU, and a maximum memory bandwidth of 400 GB/s. This configuration provided a robust platform for evaluating the performance of various models.

The number of epochs was set to three for all experiments conducted (refer to Tables 5 and 6).

Table 5. Performance metrics for monolingual subtask.

Model	Dataset	Acc (%)	P (%)	R (%)	F-Score (%)	Model Runtime (min.)
Baseline	-	74.00				
RoBERTa-large	M4	**83.00**	**84.00**	**83.00**	**83.50**	607
	AICrowd	82.50	83.50	82.00	82.74	
RoBERTa-base	M4	81.00	83.00	81.00	81.99	166
	AICrowd	79.50	80.00	78.00	78.99	
BERT-base	M4	71.00	74.00	71.00	72.47	162
	AICrowd	68.00	70.00	67.00	68.47	
DistilBERT-base-uncased	M4	68.00	73.00	68.00	70.41	77
	AICrowd	65.00	67.00	64.00	65.47	

Table 6. Performance metrics for multilingual subtask.

Model	Dataset	Acc (%)	P (%)	R (%)	F-Score (%)	Model Runtime (min.)
Baseline	-	69.00				
XLM-RoBERTa-base	Indonesian Hoax News Detection	68.00	70.00	68.00	68.99	522
	TURNBACKHOAX Dataset	67.50	69.00	67.00	67.99	
BERT-base-multilingual-cased	Indonesian Hoax News Detection	63.00	68.00	64.00	65.94	415
DistilBERT-base-multilingual-uncased	Indonesian Hoax News Detection	70.00	71.00	71.00	71.00	203
	TURNBACKHOAX Dataset	69.00	70.00	68.00	68.99	
DistilBERT-base-multilingual	ERH Corpus	**72.00**	**73.00**	**72.00**	**72.50**	400

Table 5 presents the performance metrics for the monolingual subtask, evaluated across multiple datasets including M4 and AICrowd. The metrics include accuracy (Acc), precision (P), recall (R), F-score (F-score), and model runtime (in minutes). These results highlight important trade-offs between model complexity, performance, and computational efficiency.

RoBERTa-large stands out with the highest F-score and accuracy across both datasets, achieving 83.00% accuracy and an impressive 83.50% F-score on the M4 dataset. This model's superior performance underscores its effectiveness in distinguishing between human-written and AI-generated text. The higher accuracy and F-score are indicative of its robust capacity to capture and represent nuanced textual features, making it a powerful tool for this task. However, this performance comes with a trade-off in computational cost, as reflected in its runtime of 607 min, which may be a consideration in resource-constrained environments.

RoBERTa-base presents a balanced alternative, offering a commendable performance with 81.00% accuracy and an 81.99% F-score on the M4 dataset. Its computational efficiency is also notable, with a significantly shorter runtime of 166 min compared to RoBERTa-large. This makes RoBERTa-base a viable option when a balance between performance and resource use is required.

BERT-base and **DistilBERT-base-uncased** show lower performance metrics relative to RoBERTa models. **BERT-base** achieves 71.00% accuracy and a 72.47% F-score on the M4 dataset, while **DistilBERT-base-uncased** performs slightly worse with 68.00% accuracy and a 68.47% F-score. Despite their lower performance, these models offer significant advantages in terms of computational efficiency. **DistilBERT-base-uncased** in particular, with a runtime of just 77 min, is highly resource-efficient, making it suitable for scenarios where computational resources are limited.

The performance variations between the datasets (M4 and AICrowd) highlight the importance of dataset diversity in evaluating model robustness. Different datasets can present varied linguistic characteristics and complexities, which can impact model performance. The consistent trends observed across the datasets for each model reinforce their relative strengths and limitations, providing valuable insights for selecting the most appropriate model based on specific requirements and constraints.

In conclusion, while RoBERTa-large delivers the best performance in terms of accuracy and F-score, the choice of model should consider the trade-off between performance and computational resources. Models like RoBERTa-base and DistilBERT-base-uncased offer practical alternatives depending on the computational budget and required performance level.

Table 6 outlines the performance metrics for the multilingual subtask, evaluated using datasets such as the multilingual variants from M4 and AICrowd. The metrics include accuracy (Acc), precision (P), recall (R), F-score (F-score), and model runtime (in minutes). These results shed light on the efficiency and effectiveness of the tested models in managing multilingual data.

DistilBERT-base-multilingual-cased demonstrates superior performance across both datasets, achieving an accuracy of 70.00% and an F-score of 71.00% on the M4 dataset. This model excels in handling diverse languages, reflecting its effectiveness in multilingual scenarios. Its runtime of 203 min is relatively efficient, making it a compelling choice when a balance of performance and computational resource utilization is needed. **XLM-RoBERTa-base** delivers competitive results with an accuracy of 68.00% and an F-score of 68.99%. However, it has a longer runtime of 522 min, which may be a drawback in resource-constrained environments. Despite this, its performance demonstrates robust capabilities in multilingual contexts, making it a strong contender for tasks requiring high language diversity.
BERT-base-multilingual-cased exhibits lower performance than DistilBERT and XLM-RoBERTa, achieving an accuracy of 63.00% and an F-score of 65.94%. Additionally, it has a moderate runtime of 415 min. Although it delivers acceptable results, the findings indicate that it is less effective than the other models tested in the multilingual subtask.

The performance variations across the multilingual datasets highlight the importance of selecting models tailored to specific multilingual contexts. The results indicate that models such as DistilBERT-base-multilingual-cased and XLM-RoBERTa-base are more adept at managing the complexities of multilingual text compared to BERT-base-multilingual-cased.

In summary, the multilingual task presents additional challenges compared to monolingual tasks, with varying results based on the models' ability to handle multiple languages. The inclusion of the ERH corpus in both the training and testing datasets is expected to enhance the model's generalizability by expanding the multilingual coverage. However, a clearer comparison of results with and without the ERH corpus in the training set is necessary to substantiate this claim. While the addition of the ERH corpus in the test dataset provides a broader range of languages and contexts for evaluation, its direct effect on generalizability needs to be explicitly quantified. Future experiments should compare performance metrics on training sets with and without the ERH corpus to provide a more comprehensive understanding of its impact on model generalizability.

5. Discussion

The experiments in this study evaluated the performance of various pretrained models from the BERT family, fine-tuned with a custom classifier, to detect AI-generated text.

- For **monolingual models**, the results from monolingual models offered significant insights, particularly highlighting the performance of the RoBERTa-large model. When

paired with a custom classification layer, RoBERTa-large outperformed all other models, achieving the highest accuracy and surpassing baseline results, such as those from SemEval-2024 Task 8 [58]. While DistilBERT exhibited slightly lower accuracy, it demonstrated exceptional resource efficiency, making it a practical choice for scenarios requiring computational efficiency. The RoBERTa-base model presented a strong balance between performance and training speed, delivering results comparable to RoBERTa-large but with significantly faster training times. One noteworthy experiment involved a hybrid model that combined a pretrained transformer with DistilBERT alongside a custom classifier. Although this hybrid model achieved a lower overall accuracy of 68%, it demonstrated commendable precision (73%) and exceptional resource efficiency, highlighting its potential in resource-constrained environments.

- **For multilingual models**. The multilingual presented unique challenges, including increased dataset complexity, which led to lower accuracy levels and longer training times compared to monolingual models. Interestingly, the **DistilBERT-base-multilingual-cased model** outperformed its teacher model, **BERT-base-multilingual-cased**, achieving an accuracy of 70% compared to the baseline of 68%. This finding underscores the potential of lighter models like DistilBERT in handling complex multilingual tasks while delivering competitive performance.

The findings emphasize the inherent difficulties in distinguishing between human-written and AI-generated text. While pretrained transformer models have demonstrated significant potential, they also highlight areas that require further investigation to enhance model robustness and real-world applicability.

Future research should focus on the following:

- Exploring alternative architectures, such as ALBERT (A Lite BERT for Self-supervised Learning of Language Representations) [59], for increased efficiency and scalability.
- Refining feature engineering techniques to better capture nuanced distinctions in generated text.
- Experimenting with hybrid machine learning methods to develop more sophisticated systems for diverse applications.

6. Conclusions

This study offers valuable insights into the effectiveness of transformer models, specifically BERT, RoBERTa, and DistilBERT, in detecting AI-generated text. While the results are encouraging, they also reveal persistent challenges in distinguishing between machine-generated and human-written content, particularly when handling unseen data during training. Moving forward, research should explore alternative approaches, such as A Lite BERT for Self-supervised Learning of Language Representations (ALBERT) [59], to potentially enhance model efficiency and performance.

Additionally, integrating more sophisticated feature engineering techniques and hybrid modeling approaches holds promise for overcoming current limitations. By addressing these challenges, future research can contribute to the development of more resilient AI-generated text detection systems, capable of performing effectively across diverse and challenging datasets. These advancements will be critical in ensuring the robustness and reliability of such systems for real-world applications.

Author Contributions: Conceptualization, D.G.; methodology, D.G. and C.S.-V.; software, C.S.-V.; validation, D.G. and C.S.-V.; formal analysis, D.G.; investigation, D.G.; resources, D.G.; data curation, C.S.-V.; writing—original draft preparation, D.G.; writing—review and editing, D.G.; visualization, D.G.; supervision, D.G.; project administration, D.G.; funding acquisition, D.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: As open-sources, the following datasets are released (https://paperswithcode.com/datasets; https://www.aicrowd.com/challenges/kiit-ai-mini-blitz/problems/fake-news-detection; INDONESIAN HOAX NEWS DETECTION DATASET—Mendeley Data; https://github.com/jibranfawaid/turnbackhoax-dataset/tree/main?tab=readme-ov-file#turnbackhoax-dataset, accessed on 22 July 2024), and the codebase (SemEval2024Task8/subtaskA/detector.py at main · SilviuCovaci/SemEval2024Task8 · GitHub, accessed on 22 July 2024). Additional information is available on request from the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Accuracy
ADAM	Adaptive Moment Estimation
AI	Artificial Intelligence
ALBERT	A Lite BERT for Self-supervised Learning of Language Representations
BoW	Bag of Words
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long-Short Term Memory
BLOOM	Big Science Large Open-science Open-access Multilingual Language Model
BPE	Byte Pair Encoding
CNN	Convolutional Neural Networks
DTs	Decision Trees
DL	Deep Learning
DistilBERT	Distilled BERT
GPT	Generative Pre-trained Transformer
JSON	JavaScript Object Notation
JSONL	JSON Lines
LLMs	Large Language Models
LSTM	Long Short-Term Memory
MLM	Masked Language Model
ML	Machine Learning
MT	Machine Translation
NLP	Natural Language Processing
NSP	Next Sentence Prediction
P	Precision
QA	Question Answering
R	Recall
ReLU	Rectified Linear Unit
RoBERTa	Robustly Optimized BERT
SVMs	Support Vector Machines
TC	Text Classification
TG	Text Generation
TS	Text Summarization
word2vec	Word to Vectors

References

1. Macko, D.; Moro, R.; Uchendu, A.; Lucas, J.; Yamashita, M.; Pikuliak, M.; Srba, I.; Le, T.; Lee, D.; Simko, J.; et al. MULTITuDE: Large-Scale Multilingual Machine-Generated Text Detection Benchmark. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–10 December 2023; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 9960–9987.

2. Weidinger, L.; Mellor, J.F.; Rauh, M.; Griffin, C.; Uesato, J.; Huang, P.; Cheng, M.; Glaese, M.; Balle, B.; Kasirzadeh, A.; et al. Ethical and social risks of harm from Language Models. *arXiv* **2021**, arXiv:2112.04359.
3. Solaiman, I.; Brundage, M.; Clark, J.; Askell, A.; Herbert-Voss, A.; Wu, J.; Radford, A.; Krueger, G.; Kim, J.W.; Kreps, S.; et al. Release Strategies and the Social Impacts of Language Models. *arXiv* **2019**, arXiv:1908.09203.
4. Gîfu, D. An Intelligent System for Detecting Fake News. *Procedia Comput. Sci.* **2023**, *221*, 1058–1065. [CrossRef]
5. Ermurachi, V.; Gîfu, D. UAIC1860 at SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In Proceedings of the Fourteenth Workshop on Semantic Evaluation, (SemEval-2020), Barcelona, Spain, 12–13 December 2020; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 1835–1840.
6. Gîfu, D. Utilization of Technologies for Linguistic Processing in an Electoral Context: Method LIWC-2007. In Proceedings of the Communication, Context, Interdisciplinarity Congress, Târgu-Mureș, Romania, 19–20 November 2010; "Petru Maior" University Publishing House: Târgu Mureș, Romania, 2010; Volume 1, pp. 87–98.
7. Ma, Y.; Liu, J.; Yi, F.; Cheng, Q.; Huang, Y.; Lu, W.; Liu, X. AI vs. Human-Differentiation Analysis of Scientific Content Generation. *arXiv* **2023**, arXiv:2301.10416.
8. Ouatu, B.; Gîfu, D. Chatbot, the Future of Learning? In *Ludic, Co-design and Tools Supporting Smart Learning Ecosystems and Smart Education*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 263–268.
9. Wang, Y.; Mansurov, J.; Ivanov, P.; Su, J.; Shelmanov, A.; Tsvigun, A.; Whitehouse, C.; Afzal, O.M.; Mahmoud, T.; Sasaki, T.; et al. M4: Multi-Generator, Multi-Domain, and Multi-Lingual Black-Box Machine-Generated Text Detection. *arXiv* **2023**, arXiv:2305.14902.
10. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019.
11. Manoleasa, T.; Sandu, I.; Gîfu, D.; Trandabăț, D. FII UAIC at SemEval-2022 Task 6: iSarcasmEval—Intended Sarcasm Detection in English and Arabic. In Proceedings of the 16th International Workshop on Semantic Evaluation, (SemEval-2022), Seattle, WA, USA, 14–15 July 2022; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 970–977.
12. Alexa, L.; Lorenț, A.; Gîfu, D.; Trandabăț, D. The Dabblers at SemEval-2018 Task 2: Multilingual Emoji Prediction. In Proceedings of the 12th International Workshop on Semantic Evaluation, (SemEval-2018), New Orleans, LA, USA, 5–6 June 2018; Human Language Technologies (NAACL HLT 2018). Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 405–409.
13. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2017; Volume 30.
14. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2020.
15. Niu, P.; Zhou, T.; Wang, X.; Sun, L.; Jin, R. Attention as Robust Representation for Time Series Forecasting. *arXiv* **2024**, arXiv:2402.05370v1.
16. Clark, E.; August, T.; Serrano, S.; Haduong, N.; Gururangan, S.; Smith, N.A. All That's 'Human' Is Not Gold: Evaluating Human Evaluation of Generated Text. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Bangkok, Thailand, 1–6 August 2021.
17. Gifu, D. Lexical Semantics in Text Processing. Contrastive Diachronic Studies on Romanian Language. Ph.D. Thesis, "Alexandru Ioan Cuza" University of Iași, Iași, Romania, 2016.
18. Gîfu, D. Contrastive Diachronic Study on Romanian Language. In *Proceedings FOI-2015*; Cojocaru, S., Gaindric, C., Eds.; Institute of Mathematics and Computer Science, Academy of Sciences of Moldova: Chișinău, Moldova, 2015; pp. 296–310.
19. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Edouard, G.; Myle, O.; Luke, Z.; Stoyanov, V. Unsupervised Cross-lingual Representation Learning at Scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 8440–8451.
20. Yoon, K. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014; pp. 1746–1751.
21. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
22. Garib, A.; Coffelt, T.A. DETECTing the Anomalies: Exploring Implications of Qualitative Research in Identifying AI-generated Text for AI-assisted Composition Instruction. *Comput. Compos.* **2024**, *73*, 102869. [CrossRef]
23. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. *arXiv* **2014**, arXiv:1409.3215. [CrossRef]
24. Lin, Y.; Ruan, T.; Liu, J.; Wang, H. A Survey on Neural Data-to-Text Generation. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 1431–1449. [CrossRef]

25. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
26. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
27. Liu, Y.; Lapata, M. Text Summarization with Pretrained Encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 3730–3740.
28. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. *arXiv* **2020**, arXiv:1910.01108.
29. Tang, G.; Sennrich, R.; Nivre, J. An Analysis of Attention Mechanisms: The Case of Word Sense Disambiguation in Neural Machine Translation. In Proceedings of the Third Conference on Machine Translation: Research Papers, Belgium, Brussels, 31 October–1 November 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 26–35.
30. Rajpurkar, P.; Zhang, J.; Lopyrev, K.; Liang, P. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 2383–2392. [CrossRef]
31. Yang, Y.; Cer, D.; Ahmad, A.; Guo, M.; Law, J.; Constant, N.; Abrego, G.H.; Yuan, S.; Tar, C.; Sung, Y.-H.; et al. Multilingual universal sentence encoder for semantic retrieval. *arXiv* **2019**, arXiv:1907.04307.
32. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
33. Zhang, H.; Cai, J.; Xu, J.; Wang, J. Pretraining-Based Natural Language Generation for Text Summarization. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), Hong Kong, China, 21 November 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 789–797.
34. Sun, C.; Qiu, X.; Xu, Y.; Huang, X. How to Fine-Tune BERT for Text Classification? In Proceedings of the China National Conference on Chinese Computational Linguistics, Kunming, China, 18–20 October 2019; Springer: Cham, Switzerland, 2019; pp. 194–206.
35. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.J.; Hovy, E. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS 2020), Online, 6–12 December 2020; pp. 5754–5764. Available online: https://arxiv.org/abs/1906.08237 (accessed on 22 July 2024).
36. Nguyen, D.; Naing, K.M.N.; Joshi, A. Stacking the Odds: Transformer-Based Ensemble for AI-Generated Text Detection. In Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association, Melbourne, Australia, 29 November–1 December 2023; Association for Computational Linguistics: Stroudsburg, PA, USA, 2023; pp. 173–178.
37. Garcia, J.F.; Segura-Bedmar, I. Human After All: Using Transformer Based Models to Identify Automatically Generated Text. In Proceedings of the IberLEF 2024, Valladolid, Spain, 24 September 2024.
38. Zecong, W.; Jiaxi, C.; Chen, C.; Chenhao, Y. Implementing BERT and Fine-Tuned RobertA to Detect AI Generated News by ChatGPT. *arXiv* **2023**, arXiv:2306.07401.
39. Zhang, H.; Shafig, M.O. Survey of Transformers and Towards Ensemble Learning Using Transformers for Natural Language Processing. *J. Big Data* **2024**, *11*, 25. [CrossRef] [PubMed]
40. Li, Y.; Li, Q.; Cui, L.; Bi, W.; Wang, Z.; Wang, L.; Yang, L.; Shi, S.; Zhang, Y. MAGE: Machine-generated Text Detection in the Wild. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Bangkok, Thailand, 11–16 August 2024; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 36–53.
41. Sennrich, R.; Haddow, B.; Birch, A. Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016), Berlin, Germany, 7–12 August 2016; pp. 1715–1725. [CrossRef]
42. Gage, P. A New Algorithm for Data Compression. *C Users J.* **1994**, *12*, 23–38. Available online: https://typeset.io/papers/a-new-algorithm-for-data-compression-3htk4tchd5 (accessed on 22 July 2024).
43. Raffel, C.; Shinn, C.; Roberts, A. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67. Available online: https://arxiv.org/abs/1910.10683 (accessed on 22 July 2024).
44. Smith, J.; Lee, K.; Kumar, A. Optimizing Transformer Models for Mobile Devices: A Comparative Study. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021), Punta, Cana, 7–11 November 2021; pp. 234–245. [CrossRef]
45. Koupaee, M.; Wang, W.Y. Wikihow: A Large-Scale Text Summarization Dataset. *arXiv* **2018**, arXiv:1810.09305. [CrossRef]
46. Kang, D.; Ammar, W.; Dalvi, B.; van Zuylen, M.; Kohlmeier, S.; Hovy, E.; Schwartz, R. A Dataset of Peer Reviews (PeerRead): Collection, Insights and NLP Applications. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 1647–1661.

47. Shamardina, T.; Mikhailov, V.; Cherniavskii, D.; Fenogenova, A.; Saidov, M.; Valeeva, A.; Shavrina, T.; Smurov, I.; Tutubalina, E.; Artemova, E. Findings of the Ruatd Shared Task 2022 on Artificial Text Detection in Russian. *arXiv* **2022**, arXiv:2206.01583. [CrossRef]
48. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. Llama: Open and Efficient Foundation Language Models. *arXiv* **2023**, arXiv:2302.13971.
49. Chung, H.W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, E.; Wang, X.; Dehghani, M.; Brahma, S.; et al. Scaling Instruction-Finetuned Language Models. *arXiv* **2022**, arXiv:2210.11416.
50. Muennighoff, N.; Wang, T.; Sutawika, L.; Roberts, A.; Biderman, S.; Scao, T.L.; Bari, M.S.; Shen, S.; Yong, Z.X.; Schoelkopf, H.; et al. Crosslingual Generalization Through Multitask Finetuning. *arXiv* **2022**, arXiv:2211.01786.
51. Faisal, R.; Inggrid, Y.; Rosa, A.A. Indonesian Hoax News Detection Dataset. *Mendeley Data*. 2018. Available online: https://data.mendeley.com/datasets/p3hfgr5j3m/1 (accessed on 22 July 2024).
52. Macko, D.; Moro, R.; Uchendu, A.; Lucas, J.S.; Yamashita, M.; Pikuliak, M.; Srba, I.; Le, T.; Lee, D.; Simko, J.; et al. MULTITuDE: Large-Scale Multilingual Machine-Generated Text Detection Benchmark. *arXiv* **2023**, arXiv:2310.13606.
53. McCann, B.; Bradbury, J.; Xiong, C.; Socher, R. Learned in Translation: Contextualized Word Vectors. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: New York, NY, USA, 2017; pp. 6294–6305.
54. Howard, J.; Ruder, S. Universal Language Model Fine-Tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers, Melbourne, Australia, 15–20 July 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 328–339.
55. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Huggingface's Transformers: State-of-the-Art Natural Language Processing. *arXiv* **2019**, arXiv:1910.03771.
56. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
57. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv* **2017**, arXiv:1711.05101.
58. Tran, H.T.H.; Nguyen, T.N.; Doucet, A.; Pollak, S. L3i++ at SemEval-2024 Task 8: Can Fine-tuned Large Language Model Detect Multigenerator, Multidomain, and Multilingual Black-Box Machine-Generated Text? In Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Mexico City, Mexico, 16–21 June 2024; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 13–21.
59. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations. *arXiv* **2020**, arXiv:1909.11942v6.