Plagiarism Detection Model Using Transformers

Vaishnavi Kukkala
Department of Data Science
University of New Haven
vkukk2@unh.newhaven.edu

Hari Krishna Para
Department of Data Science
University of New Haven
hpara2@unh.newhaven.edu

Sai Charan Chandu Patla
Department of Data Science
University of New Haven
schan30@unh.newhaven.edu

Abstract

Plagiarism detection is a serious difficulty in academic and professional content creation, demanding sophisticated algorithms for accurately detecting textual similarities. It is also considered serious misconduct when other professionals provide content. People tend to duplicate content from many sources and present it as their own, and detecting these plagiarized works can be difficult at times. This paper investigates some of the transformer models that can be used to detect plagiarized content. We investigate multiple models, including BERT, RoBERTa, and T5, as well as a hybrid BERT+BiLSTM architecture, to determine their performance in detecting plagiarized content. Models were fine-tuned and tested on two independent datasets (SNLI and MRPC), yielding varying results. BERT outperformed with accuracy of 85.87% (D1) and 83.77% (D2), while the BERT+LSTM hybrid achieved 85.20% accuracy on D1. RoBERTa demonstrated accuracy of 84.46% (D1) and 82.03% (D2). T5, while competitive, fell short, with accuracies of 77.84% (D1) and 66.49% (D2). These findings demonstrate the effectiveness of transformer-based techniques, specifically BERT and RoBERTa, in offering robust solutions for plagiarism detection via detailed semantic and syntactic analysis.

Keywords: *Plagiarism, deep learning, BERT, RoBERTa, T5, BiLSTM, NLP*

1. INTRODUCTION

Plagiarism is defined as using someone else's work, ideas, or creations without proper credit. This action puts someone's employment at risk while also undermining the authenticity, integrity, and legitimacy of scholarly activities. This problem has become widespread, and if not addressed in a timely manner, it will have far-reaching consequences for the global academic situation, as academic works proliferate rapidly over the world.

Plagiarism has evolved beyond mere verbatim copying, incorporating sophisticated techniques such as paraphrasing and semantic rewording, making traditional detection tools inadequate. The global spread of digital technologies and academic content exacerbates this issue, challenging institutions to preserve academic integrity and ensure fair evaluation. This necessitates innovative solutions that go beyond surface-level textual similarities and delve into deeper semantic understanding.

Plagiarism detection has become a pressing concern in academic, professional, and creative industries, where originality of content is critical. In educational institutions, the increasing availability of digital content has made it easier for students to replicate information without proper attribution, leading to a surge in academic misconduct cases. Similarly, in professional and creative domains, identifying copied or reused content is essential to uphold intellectual property rights and maintain credibility. Traditional plagiarism detection methods, primarily reliant on string matching and heuristic-based techniques, often fail to capture nuanced similarities involving paraphrasing, synonym substitutions, and structural reorganization.

In recent years, advancements in machine learning, particularly the emergence of deep learning and transformer-based models, have significantly improved the ability to understand and analyze text at a semantic level. Models such as Bidirectional Encoder Representations from Transformers (BERT), Robustly Optimized BERT Approach (RoBERTa), and Text-to-Text Transfer Transformer (T5) have demonstrated exceptional capabilities in natural language understanding tasks. These models, pre-trained on extensive corpora, can be fine-tuned for domain-specific tasks, making them ideal candidates for plagiarism detection. Additionally, hybrid architectures, such as combining BERT with Long Short-Term Memory (LSTM) networks, offer an enhanced ability to capture both contextual relationships and sequential patterns in text. These transformer-based models have revolutionized natural language processing (NLP) by their ability to understand contextual and semantic nuances in text. These models have been widely adopted across various NLP applications, including sentiment analysis, machine translation, and text summarization. Their ability to encode bidirectional context and learn rich representations of text positions them as powerful tools for detecting complex cases of plagiarism, where the relationship between original and plagiarized content is subtle.

While these models excel in many NLP tasks, their application to plagiarism detection introduces unique challenges. These include handling large volumes of data, distinguishing between legitimate common knowledge and plagiarized content, and ensuring the scalability of solutions for multilingual and domain-specific applications. Furthermore, adapting these models to the specific requirements of plagiarism detection, such as identifying non-obvious

paraphrased content, necessitates significant fine-tuning and optimization.

Hybrid approaches combining transformers with traditional machine learning techniques offer a promising avenue for tackling these challenges. For instance, BERT+LSTM architectures merge the contextual understanding of transformers with the sequential pattern recognition capabilities of LSTMs, providing a balanced framework for detecting textual similarities at multiple levels. Additionally, advancements in unsupervised learning and data augmentation are further enabling models to generalize better across diverse datasets, improving their effectiveness in real-world scenarios.

Beyond academia, plagiarism detection has broader implications for intellectual property protection, ethical journalism, and content originality in creative industries. As more industries transition to digital platforms, safeguarding originality becomes not only an ethical mandate but also a competitive necessity. This highlights the importance of research in this domain, as it contributes to the development of tools that uphold integrity and foster innovation.

Despite the significant progress, certain limitations persist. High computational demands, difficulties in handling low-resource languages, and the inherent biases in pre-trained models pose challenges to the widespread adoption of these systems. Addressing these issues requires collaborative efforts between researchers and developers, leveraging advances in hardware acceleration and inclusive dataset curation to democratize access to robust plagiarism detection tools.

This paper investigates the application of these state-of-the-art transformer-based models, including T5, RoBERTa, and BERT, along with the BERT+LSTM hybrid, for detecting plagiarized content. Using datasets specifically designed for semantic analysis, the models are evaluated across multiple performance metrics, including accuracy, recall, precision, and F1-score. The results provide insights into the comparative strengths and limitations of these approaches, highlighting their potential for robust and scalable plagiarism detection systems.

The remainder of this paper is organized as follows: Section 2 discusses related work in plagiarism detection. Section 3 outlines the proposed methodology and dataset preparation. Section 4 presents the experimental results and their analysis, followed by a discussion in Section 5. Finally, Section 6 concludes the paper with potential directions for future research.

2. LITERATURE REVIEW

Plagiarism detection in academic textual content has been a big issue in recent academia, as the majority of the content that students claim is their own is actually pirated work from other scholars. In the past, it was extremely difficult to detect academic dishonesty such as plagiarism because no solid remedy had been devised to fully address this problem.

People are more inclined than ever to plagiarize their work because of the proliferation of online information. An automated solution that will just run the textual content and identify the copied passages is required to address this issue. This study primarily explains how deep learning and natural processing language techniques can be used to construct an automated system. This issue is resolved by transformers, which guarantee the credibility and integrity of the literary works.

Multiple previous works have been conducted by different authors in relation to plagiarism detection using different techniques.

In the paper 1, "**A Novel Plagiarism Detection Approach Combining BERT-based Word Embedding, Attention-based LSTMs, and an Improved Differential Evolution Algorithm**," the difficulty of identifying complex plagiarism in unbalanced datasets—such as paraphrasing and semantic similarity—is discussed. By combining attention-based LSTMs to capture contextual dependencies, a differential evolution (DE) method to optimize training, and BERT for word embeddings, the authors suggest a hybrid model. The class imbalance problem is addressed by a focal loss function. When tested on datasets such as MSRP and SNLI, the strategy outperforms traditional and population-based approaches. Limitations, however, include the need for domain-specific fine-tuning to generalize across various text forms and the computational intensity.

The paper 2. "A Comprehensive Approach to Plagiarism Detection Using BERT and Multilingual Models" tackles the problem of recognizing information that has been paraphrased and is semantically comparable across languages. To enhance cross-lingual plagiarism detection, the authors suggest a pipeline that makes use of BERT and multilingual transformer models such as XLM-R. To improve model accuracy, their method incorporates domain-specific fine-tuning, semantic similarity measures, and phrase embeddings. Using benchmark datasets, the study shows how effective transformers are in identifying subtle forms of plagiarism with competitive accuracy. However, it draws attention to drawbacks that may impede scalability and performance in a variety of real-world scenarios, including high computing costs and the requirement for large amounts of labeled data for fine-tuning.

A sophisticated NLP-based plagiarism detection model that outperforms conventional exact match algorithms is presented by the author in paper 3. It can distinguish between literal and paraphrased information by using methods like

AHP, linguistic complexity, and semantic analysis. When tested against both reference and suspect documents, the model's accuracy was 72.5% at Level 1 and 80% at Level 2. The study emphasizes its value for research and academic integrity, offering a practical tool for institutions and teachers. The model's scalability and domain-specific adaptability, however, could be investigated further.

In order to identify plagiarism, paper 4 suggests a Siamese network model that combines BERT and Bi-LSTM. It uses multi-head self-attention and a semantic interaction mechanism to increase accuracy. The large-scale BERT model may be computationally demanding, and fine-tuning requires a lot of labelled data, which is one of the model's drawbacks despite its efficacy. Future research will focus on improving its use in teaching.

The paper 5. "NLP-based Deep Learning Approach for Plagiarism Detection" introduces a new method of detecting plagiarism that makes use of deep learning models, namely BERT and Bi-LSTM. It presents an advanced technique that successfully detects both direct and paraphrased plagiarism by fusing semantic analysis, contextual awareness, and deep learning capacity. The model employs an attention strategy for semantic alignment after using BERT for initial text encoding and Bi-LSTM for contextual feature extraction and interaction. Compared to conventional techniques, this strategy greatly increases the effectiveness of plagiarism detection. The model's rich semantic comprehension improves plagiarism detection and can be a helpful tool in academic and professional settings, according to the scientists' conclusion.

Nonetheless, the study notes a number of drawbacks, such as the computational difficulty of deep model training, which necessitates significant resources, and the reliance on extensive labelled datasets for efficient training. Furthermore, even though the model works well overall, it is still difficult to adjust and modify for domain-specific texts.

The study "Deep Learning Detection Method for Large Language Models-Generated Scientific Content" investigates the application of sophisticated deep learning methods to recognize scientific writing produced by artificial intelligence. Semantic analysis and transformer-based models are used to improve identification. The suggested method's ability to guarantee academic integrity is demonstrated by its excellent accuracy in differentiating between content created by AI and that written by humans. Limitations include the difficulty of responding to different linguistic styles and the high computational cost of training big models (paper 6).

The paper 7. explores advanced NLP techniques for detecting plagiarism, emphasizing semantic analysis to identify paraphrasing and semantic similarity. The approach integrates syntactic and semantic representations using transformers and attention mechanisms. The limitations include high computational cost and challenges in real-time application scalability.

In article 8, the authors address literal and semantic plagiarism by combining machine learning and natural language processing. It presents scalable solutions for large-scale applications by investigating methods for identifying contextual similarities and paraphrasing in multilingual datasets. Computational complexity and cross-domain generalization are obstacles.

Innovative methods for identifying plagiarism in programming assignments are covered in paper 9, "Plagiarism Detection in Source Code using Machine Learning and NLP Techniques." It employs NLP and ML techniques to find structural and semantic parallels in source code, with a focus on tokenization and syntactic analysis to efficiently capture obfuscations and changes. Even when using obfuscation techniques, the results show great accuracy in identifying plagiarism; nonetheless, there are certain limitations, such as difficulties with scalability for huge datasets and managing different programming styles.

The paper 10, The paper titled "How Large Language Models are Transforming Machine-Paraphrased Plagiarism" (2022) explores the impact of large language models (LLMs) like GPT on the evolution of paraphrased plagiarism. It looks at the difficulties that these models present for conventional plagiarism detection algorithms because of their capacity to produce content that is cohesive and semantically comparable. For increased efficacy, the study suggests adaptive detection techniques that make use of transformer designs. Computational expenses and the requirement for constant adjustment to changing model capabilities are among the limitations.

3. RESEARCH METHODOLOGY

3.1 Introduction

A thorough natural language processing (NLP) pipeline is used in this work to assess how well transformer-based models and hybrid architectures identify plagiarism. The approach is intended to preprocess textual material, optimize language models that have already been trained, and assess the models' performance using a variety of metrics. The pipeline's steps are as follows:

3.2 Dataset Description

Two datasets for plagiarism detection were chosen by us: [Figure 1] SNLI stands for Stanford Natural Language Inference. The primary purpose of the SNLI dataset was natural language inference (NLI) tasks, which classify the relationship between two sentences into one of three categories: neutral, contradiction, or entailment. This dataset aims to give models the ability to comprehend different semantic links between sentence pairs. SNLI was modified for plagiarism detection in this research in order to find similar or

paraphrased text by treating relationships between sentences as indicative of plagiarism. With more than 570,152 sentence pairings, the dataset is appropriate for optimizing big transformer models like BERT, RoBERTa, and T5.

Text	Judgments	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction C C C C C	The man is sleeping
An older and younger man smiling.	neutral N N E N N	Two men are smiling and laughing at the cats playing on the floor.
A black race car starts up in front of a crowd of people.	contradiction C C C C C	A man is driving down a lonely road.
A soccer game with multiple males playing.	entailment E E E E E	Some men are playing a sport.
A smiling costumed woman is holding an umbrella.	neutral N N E C N	A happy woman in a fairy costume holds an umbrella.

Figure 1: SNLI Dataset

Figure 2 shows the Microsoft Research Paraphrase Corpus, or MRPC. The MRPC dataset is strongly related to plagiarism detection tasks since it focuses on identifying if two sentences are paraphrases of one another. In order to differentiate plagiarized text from original information, the model learns to categorize sentence pairs as either paraphrased or not using MRPC. 5,801 sentence pairings make up the dataset, with almost equal numbers of positive (paraphrased) and negative (non-paraphrased) cases. Effective training and assessment of models for detecting textual similarities in plagiarism detection are made possible by this balanced distribution.

label int64	id_1 int64	id_2 int64	sentence1 string · lengths	sentence2 string · lengths
1	702,876	702,977	Amrozi accused his brother, whom he called "the witness", of…	Referring to him as only "the witness", Amrozi accused his…
0	2,108,705	2,108,831	Yucaipa owned Dominick's before selling the chain to Safeway in…	Yucaipa bought Dominick's in 1995 for $693 million and sold it to…
1	1,330,381	1,330,521	They had published an advertisement on the Internet on…	On June 10, the ship's owners had published an advertisement on the…
0	3,344,667	3,344,648	Around 0335 GMT, Tab shares were up 19 cents, or 4.4%, at A$4.56,…	Tab shares jumped 20 cents, or 4.6%, to set a record closing hig…
1	1,236,820	1,236,712	The stock rose $2.11, or about 11 percent, to close Friday at $21.5…	PG&E Corp. shares jumped $1.63 or 8 percent to $21.03 on the New…
1	738,533	737,951	Revenue in the first quarter of the year dropped 15 percent from…	With the scandal hanging over Stewart's company, revenue the…

Figure 2: MRPC Dataset

3.3 Dataset Pre-processing

A crucial step in getting the data ready for transformer-based models like BERT, RoBERTa, and T5 is data preparation. In order to prepare the text for training and assessment, the preprocessing pipeline was created to standardize and sanitize it. It often entails a number of steps that culminate in more reliable data for additional research.

3.3.1 Dataset Preparation

The studies were conducted using two datasets: Dataset 1 (D1) - SNLI and Dataset 2 (D2) - MRPC. Each pair of short texts in Dataset 1 (D1) has been labeled as either non-plagiarized or plagiarized. To mimic actual plagiarism situations, the dataset contains examples of synonym replacement, structural reorganization, and semantic paraphrasing. Longer text passages with more intricate semantic linkages are included in Dataset 2 (D2), where several phrases from a single text are taken from various sources. A source text, a candidate text, and a binary label indicating whether the candidate text is plagiarized (1) or not (0) are included in every dataset entry.

To guarantee consistency throughout the studies, both datasets underwent preprocessing. In this preprocessing stage, the text was cleaned up by eliminating special characters, normalized to lowercase, and tokenized using model-specific tokenizers into subword units. After that, the datasets were divided into three groups: 70% for training, 15% for validation, and 15% for testing.

3.3.2 Tokenization

The process of breaking up a text into smaller chunks, like words or subwords, is called tokenization. Tokenization for this project was carried out using Hugging Face transformers library model-specific tokenizers that were tailored for every model:

- **BERT Tokenizer**: The BERT tokenizer, which divides the text into subword units (WordPiece), was used to tokenize the text. The vocabulary of the model is used to map each word or subword to a corresponding token ID.
- **RoBERTa Tokenizer**: The text was divided into subword tokens using RoBERTa's tokenizer. It uses vocabulary-based tokenization and dynamic masking, just like BERT.
- **T5 Tokenizer**: T5 tokenizes using a SentencePiece model, which can effectively handle a variety of inputs and languages.

3.3.3 Handling of Special Tokens

Special tokens were added to the tokenized text for each transformer model to signify the start of a sequence, the break between sentence pairs, or the end of the sequence:

- **BERT and RoBERTa**: The aggregate information of the text is represented by the [CLS] token at the start of the input sequence in both models. To mark the separation of the two texts, a [SEP] token is appended in between sentence pairs.
- **T5**: Task-specific tokens are incorporated into the T5 model. For example, the model's task is indicated by the prefix <plagiarism-detection>, which is followed by the input text.

  These special tokens help the model understand the structure of the input and how to process it correctly during training and inference.

3.3.4 Padding and Truncation

For transformer models to process batches efficiently, input sequences must be the same length. Consequently, the tokenized sequences were subjected to truncation and padding:

- **Padding**: To guarantee consistent length across all samples in a batch, tokenized sequences shorter than the maximum permitted length (512 tokens for the

majority of models) were padded with the [PAD] token.

- **Truncation**: To ensure that no sequence goes over the maximum token limit, sequences that were longer than the permitted maximum length were trimmed to meet the model's input needs.

In addition to avoiding memory overflow during training, padding and truncation guarantee that the input data is reliable and consistent with the transformer models.

3.3.5 Text Vectorization

The text is converted into embedding vectors following tokenization. Pre-trained models like BERT, RoBERTa, and T5 are the source of these embeddings. The meaning and contextual relationships within the text are captured by the embeddings, which represent the semantic content of the words or subwords. In the context of the full text, each token ID correlates to a dense vector that captures its meaning. The transformer models use the embeddings as input for both inference and training.

A Long Short-Term Memory (LSTM) layer is traversed by the token embeddings in the BERT+LSTM hybrid model. This extra layer enables the model to learn context over longer text sequences by capturing the sequential dependencies between characters.The LSTM layer complements the transformer model by learning from the sequence of token embeddings generated by BERT.

3.3.6 Text Preparation for Classification

The processed sequences (together with the matching attention masks) are at last prepared for input into the models following preprocessing and tokenization. For classification, each tokenized sequence is given into transformer models, which are trained to determine whether or not the text is plagiarized. Performance indicators including accuracy, precision, recall, and F1 score are then used to assess the results.

3.4 Model Selection

To identify plagiarism in text, we used a number of cutting-edge models in this study, including T5, RoBERTa, BERT, and a hybrid model called BERT+LSTM. Because of their shown effectiveness in natural language processing (NLP) tasks—specifically, text categorization, sentence similarity, and semantic understanding—all of these models were chosen. Using the prepared dataset, these models were trained by refining previously learned versions on a plagiarism detection task. A thorough description of each model and how it was trained is given below.

3.4.1 BERT (Bidirectional Encoder Representations from Transformers)

One of the most popular transformer models is BERT, which is renowned for its capacity to pre-train on vast volumes of text and refine on subsequent tasks like sentence categorization. BERT's bidirectional attention mechanism enables it to comprehend context from both left-to-right and right-to-left directions, in contrast to unidirectional models. This feature is very helpful for detecting plagiarism, as it is crucial to comprehend the connections between words and their contexts.

3.4.2 RoBERTa (Robustly Optimized BERT Approach)

An improved version of BERT, RoBERTa, was trained using longer sequences, more data, and larger batch sizes. It employs a dynamic masking strategy during training and eliminates the Next Sentence Prediction aim from BERT. It has been demonstrated that RoBERTa performs better than BERT in a variety of NLP tasks, especially those that call for a high level of contextual knowledge.

3.4.3 BERT + LSTM (Long Short-Term Memory)

To capture sequential dependencies in the text, the BERT + LSTM hybrid model combines an LSTM layer with the pre-trained BERT embeddings. The LSTM layer aids in learning the long-term dependencies and relationships between characters inside the sentence, whereas BERT records rich contextual embeddings. In order to leverage the advantages of both models—LSTM for sequential dependencies and BERT for contextual embeddings—a hybrid method was selected.

3.4.4 T5 (Text-to-Text Transfer Transformer)

Google created the transformer-based T5 paradigm, which is intended to handle a variety of natural language processing tasks by presenting them as text-to-text problems. Accordingly, T5 transforms various tasks (such as translation, summarization, and classification) into a single format: the model receives input text, and the output is likewise a text sequence. The model was trained to determine if a particular text pair reflected plagiarism or not, and we refined T5 on a binary classification task for this work.

4.0 Training and Optimization Strategy

This section now offers a thorough, in-depth description of the training procedure. As seen in Figure 3, it explains training configurations, optimization techniques, and the fine-tuning procedure.

The training process for all models was carried out with the following steps:

1. **Data Splitting**: The dataset was split into training (70%), validation (15%), and test (15%) sets. This ensured that the models were evaluated on unseen data after training.
2. **Loss Function**: The binary classification job, which aimed to determine whether or not a pair of texts was plagiarized, employed a binary cross-entropy loss.
3. **Batch Size**: In order to balance GPU memory utilization and computational efficiency, a batch size of 16 was selected for all models.
4. **Evaluation Metrics**: Accuracy, precision, recall, and F1 score on the validation set were used to monitor

performance throughout training. Early stopping was used to avoid overfitting, and the models were assessed at the conclusion of each session.

5. **Hardware**: A GPU-equipped computer was used to train the models, taking advantage of parallel processing capabilities for effective computation.

4.1 Fine-Tuning and Hyperparameter Optimization

Tuning the hyperparameters was essential to getting the best results. The following hyperparameters were adjusted for every model:

- Learning Rate: To avoid underfitting or overfitting, the learning rate was selected after experimenting with various values (1e−5, 5e−5, and 1e−4).
- Epochs: Training was performed for 5 epochs, as further training resulted in diminishing returns and potential overfitting.
- Batch Size: A batch size of 16 was selected based on hardware constraints and memory efficiency.

The best hyperparameters were selected based on the validation set performance, ensuring that the models generalized well to unseen data.

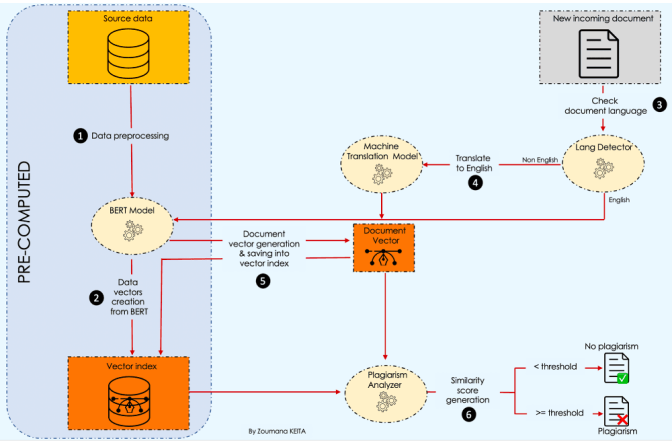

Figure 3. Flowchart of the project

5.0 Evaluation & Results

In this section, each model's results are thoroughly analyzed and evaluated to determine its performance in relation to plagiarism detection.

To evaluate the performance of each model, we used several standard metrics:

- **Accuracy**: The proportion of correctly classified instances (i.e., correctly identifying plagiarized or non-plagiarized texts).
- **Precision**: The ratio of true positive predictions to the total number of positive predictions.
- **Recall**: The ratio of true positive predictions to the total number of actual positive instances.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two.

Both datasets, D1 and D2, were used to test the models' performance under various text length and complexity levels. While D2's longer, more intricate passages test the models' ability to detect subtle or indirect copying, D1's short-text pairs test the models' ability to detect more obvious types of plagiarism.

5.1 Results

5.1.1 T5

The accuracy of the T5 model was 66.49% on the MSRP dataset (D2) and 77.84% on the SNLI dataset (D1) (Figure 4). On D2, T5 struggled with recall and precision, achieving 50.00% recall and 33.25% precision, while performing reasonably well on D1 in terms of accuracy. This suggests that T5 had trouble differentiating non-plagiarized material, resulting in a lower F1 score of 39.94% on D2 (Figure 5), even though it was able to properly categorize some plagiarized pairs.

Classification Report:

```
Accuracy: 0.7784
precision: 0.7800
recall: 0.7779
f1: 0.7776
Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.80      0.83      3368
           1       0.73      0.70      0.71      3219
           2       0.74      0.83      0.79      3237

    accuracy                           0.78      9824
   macro avg       0.78      0.78      0.78      9824
weighted avg       0.78      0.78      0.78      9824
```

Figure 4: T5 model's evaluation metrics on SNLI dataset

```
Accuracy: 0.6649
precision: 0.3325
recall: 0.5000
f1: 0.3994
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       578
           1       0.66      1.00      0.80      1147

    accuracy                           0.66      1725
   macro avg       0.33      0.50      0.40      1725
weighted avg       0.44      0.66      0.53      1725
```

Figure 5: T5 model's evaluation metrics on MRPC dataset

5.1.2 RoBERTa Model Results

When compared to T5, the RoBERTa model performed better, especially on the SNLI dataset (D1)–Figure 6. 84.46% accuracy, 84.46% recall, and 84.50% precision were attained. RoBERTa outperformed T5 on the MSRP dataset (D2) – Figure 7, achieving 82.03% accuracy, 82.03% recall, and 81.70% precision. With an F1 score of 81.67% on D2,

RoBERTa demonstrated its capacity to effectively detect plagiarized content while reducing false positives.

Classification Report:

Accuracy: 0.8446
precision: 0.8450
recall: 0.8446
f1: 0.8447
Classification Report:

	precision	recall	f1-score	support
entailment	0.88	0.86	0.87	3368
neutral	0.79	0.80	0.80	3219
contradiction	0.86	0.86	0.86	3237
accuracy			0.84	9824
macro avg	0.84	0.84	0.84	9824
weighted avg	0.84	0.84	0.84	9824

Figure 6: Roberta model's evaluation metrics on SNLI dataset

Accuracy: 0.8203
precision: 0.8170
recall: 0.8203
f1: 0.8167
Classification Report:

	precision	recall	f1-score	support
0	0.77	0.66	0.71	578
1	0.84	0.90	0.87	1147
accuracy			0.82	1725
macro avg	0.80	0.78	0.79	1725
weighted avg	0.82	0.82	0.82	1725

Figure 7: Roberta model's evaluation metrics on MRPC dataset

5.1.3 BERT Model Results

With an accuracy of 85.87% and strong recall (85.87%) and precision (85.94%) on the SNLI dataset (D1) – Figure 8, the BERT model fared better than both T5 and RoBERTa. With an F1 score of 85.90% on D1, BERT demonstrated a high level of efficacy in detecting plagiarism. BERT also demonstrated strong performance on the MSRP dataset (D2) – Figure 9, achieving 83.77% accuracy, 81.61% recall, 81.84% precision, and 81.72% F1 score. These findings show that BERT can retain high performance across all assessment parameters while generalizing well across both datasets.

Classification Report:

Accuracy: 0.8587
precision: 0.8594
recall: 0.8587
f1: 0.8590
Classification Report:

	precision	recall	f1-score	support
entailment	0.89	0.86	0.88	3368
neutral	0.81	0.83	0.82	3219
contradiction	0.88	0.88	0.88	3237
accuracy			0.86	9824
macro avg	0.86	0.86	0.86	9824
weighted avg	0.86	0.86	0.86	9824

Figure 8: BERT model's evaluation metrics on SNLI dataset

Accuracy: 0.8377
precision: 0.8184
recall: 0.8161
f1: 0.8172
Classification Report:

	precision	recall	f1-score	support
0	0.76	0.75	0.76	578
1	0.88	0.88	0.88	1147
accuracy			0.84	1725
macro avg	0.82	0.82	0.82	1725
weighted avg	0.84	0.84	0.84	1725

Figure 9: BERT model's evaluation metrics on MRPC dataset

5.1.4 BERT + LSTM Model Results

On both datasets, the BERT + LSTM hybrid model produced the best accuracy. It achieved an accuracy of 85.20%, recall of 85.20%, precision of 85.20%, and F1 score of 82.84% on the SNLI dataset (D1) - Figure 10. This implies that greater classification performance was achieved by incorporating an LSTM layer into BERT's pre-trained embeddings, which assisted in capturing sequential dependencies in text. BERT + LSTM performed well on the MSRP dataset (D2) – Figure 11, achieving 82.84% accuracy, 78.69% recall, 81.74% precision, and 79.85% F1 score.

Classification Report:

```
Accuracy: 0.8520
precision: 0.8516
recall: 0.8520
f1: 0.8520
Classification Report:
               precision    recall  f1-score   support

   entailment       0.87      0.87      0.87      3368
      neutral       0.81      0.81      0.81      3219
contradiction       0.87      0.87      0.87      3237

     accuracy                           0.85      9824
    macro avg       0.85      0.85      0.85      9824
 weighted avg       0.85      0.85      0.85      9824
```

Figure 10: BERT+LSTM model's evaluation metrics on SNLI dataset

```
Accuracy: 0.8284
precision: 0.8174
recall: 0.7869
f1: 0.7985
Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.66      0.72       578
           1       0.84      0.91      0.88      1147

    accuracy                           0.83      1725
   macro avg       0.82      0.79      0.80      1725
weighted avg       0.83      0.83      0.82      1725
```

Figure 11: BERT+LSTM model's evaluation metrics on MRPC

5.2 Comparative Analysis of the Models

This section introduces a comprehensive analysis of the four models' performance. The [Table 1] shows the models and their accuracy scores.

Model	Accuracy (D1)	Accuracy (D2)
T5	77.84%	66.49%
RoBERTa	84.46%	82.03%
BERT + LSTM	85.20%	82.84%
BERT	85.87%	83.77%

Table 1: Model performance comparison table

The findings demonstrate that, with the highest accuracy and F1 score, BERT continuously outperformed the other models on the SNLI (D1) dataset. BERT beat BERT + LSTM on the MSRP (D2) dataset, demonstrating that the hybrid model was less successful than BERT's pre-trained transformer architecture. Additionally, RoBERTa performed well, especially on D1. As seen in Figure 12, T5 had the worst results, particularly on the MSRP (D2) dataset.

1. D1 --> SNLI Dataset
2. D2 --> MSRP Dataset

Model	Accuracy (D1)	Recall (D1)	Precision (D1)	F1 Score (D1)	Accuracy (D2)	Recall (D2)	Precision (D2)	F1 Score (D2)
T5	0.7784	0.7779	0.7800	0.7776	0.6649	0.5000	0.3325	0.3994
RoBERTa	0.8446	0.8446	0.8450	0.8447	0.8203	0.8203	0.8170	0.8167
BERT	0.8587	0.8587	0.8594	0.8590	0.8377	0.8161	0.8184	0.8172
BERT + LSTM	0.8520	0.8520	0.8516	0.8520	0. 8284	0.7869	0.8174	0.7985

Figure 12: Comprehensive analysis

6.0 Conclusion

Plagiarism has been a significant concern due to the rising availability of online resources that people can easily copy without proper recognition. The main goal of this study was to create models for detecting plagiarism in texts by utilizing transformers. Both direct and paraphrased plagiarism can be successfully detected by transformer models such as BERT and RoBERTa, which are very good at detecting plagiarism. The project shows how great accuracy may be attained in these tasks using contemporary NLP approaches. Furthermore, this work establishes the foundation for next developments that can enhance the effectiveness and application of plagiarism detection systems, such as multilingual and real-time detection.

6.1 Future Work

In order to increase the accuracy of detecting plagiarism in a variety of domains, we would like to concentrate on improving transformer models using domain-specific datasets in the future. Furthermore, the models' utility in various academic and international contexts will be enhanced by adding bilingual support and real-time detection. The resilience of plagiarism detection systems could be further increased by investigating hybrid strategies that combine the advantages of transformers with rule-based systems or other similarity metrics.

6.2 Limitations

The transformer models are computationally expensive, limiting scalability in resource-constrained environments.

7 References

1. Moravvej, S. V., Mousavirad, S. J., Oliva, D., & Mohammadi, F. "A Novel Plagiarism Detection Approach Combining BERT-based Word Embedding, Attention-based LSTMs, and an Improved Differential Evolution Algorithm." arXiv preprint arXiv:2305.00347, 2023 https://arxiv.org/pdf/2305.02374
2. Mujahid Ali Quidwai, Chunhui Li, Parijat Dube, "A Comprehensive Approach to Plagiarism Detection Using BERT and Multilingual Models," *arXiv preprint arXiv:2306.08122*, 2023. [Online]. Available: https://arxiv.org/pdf/2306.08122.
3. R. Patil, V. Kadam, R. Nakate, S. Kadam, S. Pattade and M. Mitkari, "A Novel Natural Language Processing Based Model for Plagiarism

Detection," *2024 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India, 2024, pp. 1-5, doi: 10.1109/ESCI59607.2024.10497386. https://ieeexplore.ieee.org/document/10497386

4. J. Zhang, S. Xue, J. L. Li and J. She, "Automated Plagiarism Detection Model Based On Deep Siamese Network," *2022 IEEE 8th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, Chengdu, China, 2022, pp. 298-302, doi: 10.1109/CCIS57298.2022.10016354. https://ieeexplore.ieee.org/abstract/document/10016354
5. Rosu, R., Stoica, A., Popescu, P., & Mihăescu, M. (2020). NLP-based Deep Learning Approach for Plagiarism Detection. *ResearchGate*. Available: https://www.researchgate.net/publication/347682908_NLP_based_Deep_Learning_Approach_for_Plagiarism_Detection
6. Alhijawi, B., Jarrar, R., AbuAlRub, A. *et al.* Deep learning detection method for large language models-generated scientific content. *Neural Comput & Applic* (2024). https://doi.org/10.1007/s00521-024-10538-y
7. S. K. Pal, O. J. Raffik, R. Roy, V. B. Lalman, S. Srivastava and B. Sharma, "Automatic Plagiarism Detection Using Natural Language Processing," *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2023, pp. 218-222. https://ieeexplore.ieee.org/document/10112546
8. A. Vasuteja, A. V. Reddy and A. Pravin, "Beyond Copy Paste: Plagiarism Detection using Machine Learning," *2024 International Conference on Inventive Computation Technologies (ICICT)*, Lalitpur, Nepal, 2024, pp. 245-251, doi: 10.1109/ICICT60155.2024.10544470. https://ieeexplore.ieee.org/document/10544470
9. W. Ali *et al*., "A Novel Framework for Plagiarism Detection: A Case Study for Urdu Language," *2018 24th International Conference on Automation and Computing (ICAC)*, Newcastle Upon Tyne, UK, 2018, pp. 1-6, doi: 10.23919/IConAC.2018.8749122. https://ieeexplore.ieee.org/document/8749122
10. J. P. Wahle, T. Ruas, F. Kirstein, and B. Gipp, “How Large Language Models are Transforming MachineParaphrased Plagiarism”, in Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP). https://arxiv.org/pdf/2210.03568