IR Final Project Report

Supervisor: Dr. BabaAli

Aaron Bateni

Arad Vazir Panah

Nima Niroomand

Yazdan Zandiye Vakili

Introduction

This project is a multifaceted endeavor that begins with the construction of a unique dataset consisting of texts from ten different authors, each a master of their craft in the realm of Persian text composition. The dataset will be meticulously assembled to include multiple documents for each author, with each document comprising about 500 words, ensuring a structured approach to data analysis. The chosen texts will span a range of themes and styles, reflecting the rich diversity found within the corpus of Persian writing, and will include essential metadata like author names and content descriptors.

In the second phase, the project shifts focus to the challenge of author identification through the application of advanced natural language processing techniques. Here, we will harness the power of BERT language models from Hugging Face, a platform renowned for its state-of-the-art machine learning tools. The task will involve fine-tuning a BERT based model specifically for the intricate task of distinguishing between the unique textual fingerprints left by each author in the dataset. This not only serves as a testament to the capabilities of modern AI in understanding and categorizing language patterns but also stands as an innovative intersection of technology and the rich heritage of Persian textual works.

Dataset Construction

Embarking on the creation of our dataset, we intentionally steered away from automated techniques like web scraping, choosing instead a hands-on approach to selecting authors and their texts. This deliberate process, although labor-intensive, has yielded a dataset of remarkable purity and complexity. Our focus is on the mystery and supernatural genres, guided by a personal affinity for these captivating realms of fiction. We handpicked a selection of texts from the most iconic works of each author, ensuring the dataset resonates with the essence of these beloved narratives.

The assembly of our dataset was an exercise in meticulous curation. We manually gathered texts from multiple books by each of the ten authors, who, while not of Iranian origin and thus their works not natively in Persian, are nonetheless integral to our study. This manual process allowed us to build a comprehensive CSV file, pairing texts with

their respective authors and author IDs, ensuring precision and clarity as exemplified by the tables provided.

Our journey was not without its challenges. The authors' original works were not in Persian, and sourcing Persian versions, particularly free editions, proved to be a formidable hurdle. However, determined research and a bit of creative searching eventually led us to the resources we needed. By overcoming these obstacles, we have compiled a dataset that not only serves our specific needs but also stands as a testament to the dedication and rigor that underpins our project.

In the end, the dataset we have crafted is a rich tapestry of narrative artistry, one that bridges cultural and linguistic divides. It promises to be an ideal training ground for the BERT models we intend to deploy, testing the limits of AI in the domain of text analysis and author identification. Through this careful and considered compilation process, we aim to honor the depth and diversity of the genres we admire while paving the way for innovative strides in computational linguistics.

Author	Author ID	Number of Docs
Artemis Fowl	1	159
Anthony Horowitz	2	109
Brandon Malle	3	257
John Flanagan	4	129
D.G.McHale	5	173
Rick Riordan	6	228
D.B.Reynolds	7	76
J.K.ROLING	8	92
R. L. Stine	9	56
c.s.lewis	10	102

Preprocessing

To enhance the quality of our dataset for the Persian text, we employed the *'hazm'* package, a component of the *nltk* family specifically designed for Persian language processing. Our preprocessing regimen was a critical step, given that the BERT tokenizer, while robust, does not handle certain tasks such as stemming—a process crucial for reducing words to their root form to maintain consistency across different variants.

The preprocessing workflow was meticulously structured. Initially, we purged the text of any special characters to ensure a uniform textual canvas. Subsequently, we utilized *'hazm'* to eliminate Persian stop words, which are common words that add little semantic value to the analysis. This was followed by the removal of any remnants of the English language, including words and numerals, to preserve the integrity of the Persian text.

The final stages of preprocessing involved stemming and lemmatization. Stemming was successfully implemented to distill words down to their base or root form. However, upon evaluating the lemmatization capabilities of *'hazm'*, we found that it did not meet our standards of quality. Consequently, we decided to forego lemmatization in favor of relying solely on the stemmed data, which proved to be more effective for our purposes.

For those interested in the specifics of our preprocessing approach, the entire procedure has been documented and is accessible in the 'preprocessing.py' file. This file provides a transparent view of the methods we employed to curate our dataset meticulously, ensuring that it was optimally prepared for the subsequent stages of our project.

Model Selection

In the pursuit of an optimal model for our project, we evaluated several BERT-based architectures, each with its own set of trade-offs in terms of computational cost and performance metrics. Below is a detailed account of our experiences with these models:

BERT Base Multilingual:

This model proved to be computationally intensive to fine-tune, requiring approximately one hour for just three epochs, and resulted in a modest accuracy of 50%. The resource demands made it a less attractive option for our purposes.

DistilBERT:

A lighter and less resource-heavy model, DistilBERT allowed us to complete fine-tuning in about 20 minutes over three epochs. However, the model's performance was underwhelming, achieving only 33% accuracy in our evaluations.

RoBERTa (Multilingual Variant):

We considered the multilingual version of RoBERTa, which is known for its robust performance. Nevertheless, the resource constraints, even on platforms like Google Colab, hindered our ability to fine-tune this model effectively.

DistilRoBERTa:

Offering a more time-efficient fine-tuning process at roughly 15 minutes for three epochs, DistilRoBERTa fell short in evaluation, recording a mere 18% accuracy. While faster, its performance was significantly inferior compared to the other models we tested.

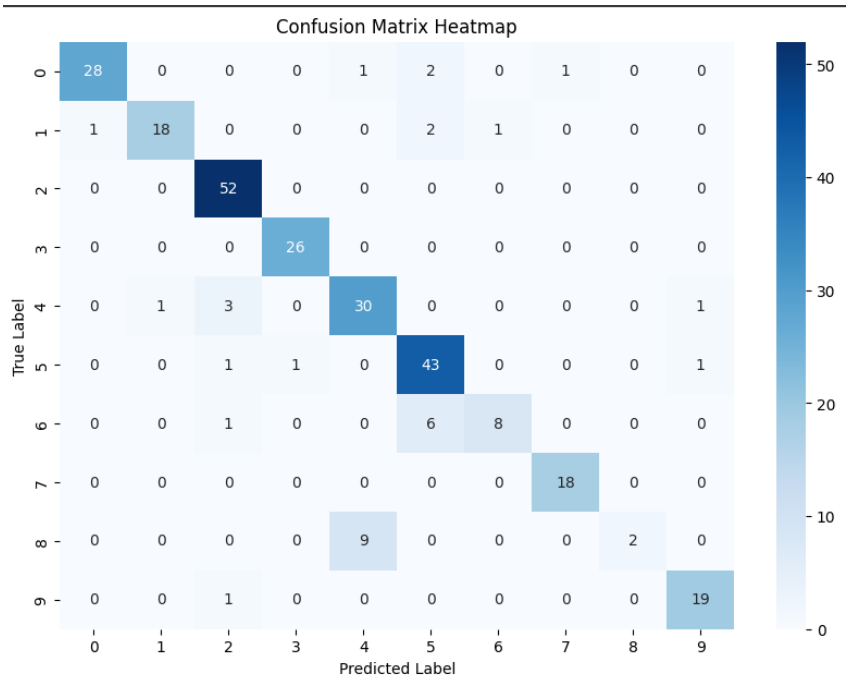
BERT Base Persian (HooshvareLab/bert-fa-base-uncased)(Main model):

Our efforts culminated in the use of a BERT Base model specifically pre-trained for Persian text. Despite the longer fine-tuning time—about an hour for three epochs and two hours for six—it delivered a promising accuracy of 65% and 88%, respectively. This model became the cornerstone of our project. We set the maximum sequence length for BERT to 128 and kept most hyperparameters at their default values, with minimal modifications.

Author ID	Precision	Recall	F1 Score	Support
1	97	88	92	32
2	95	82	88	22
3	90	100	95	52
4	96	100	98	26
5	75	86	80	35
6	81	93	87	46
7	89	53	67	15
8	95	100	97	18
9	100	18	31	11
10	90	95	93	20

Metric	Value
Accuracy	88
Precision	89
Recall	88
F1 Score	87
Support	277

Confusion Matrix



Comparatively, while **RoBERTa** is generally considered superior to **BERT** in performance, it also demands more complexity and resources. To extract improved results from RoBERTa, a longer training time with an increased number of epochs would be necessary. However, given our resource constraints and the time-accuracy trade-off, the Persian-specific BERT model emerged as the most feasible and effective for our project.

Experiment with 5 Folds Cross Validation

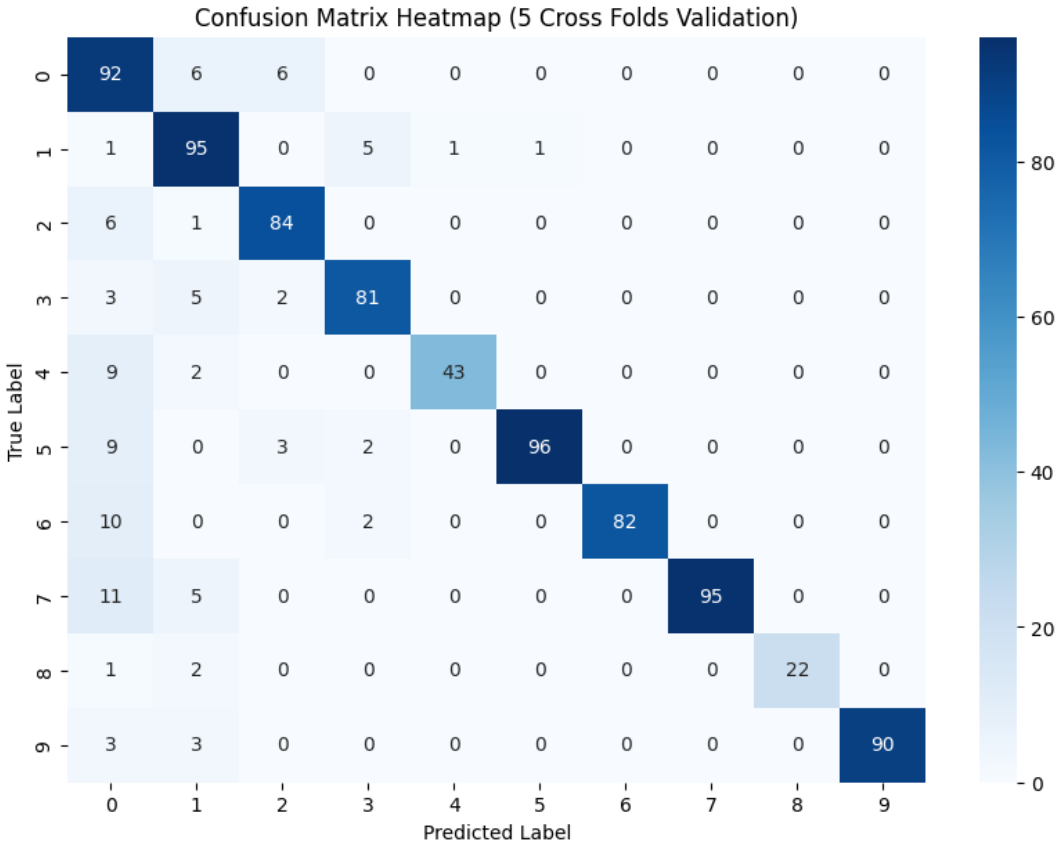
In our ongoing quest to optimize model performance, we implemented a five-fold cross-validation technique on the most promising model identified in the preceding section of our experiment. This approach yielded even more encouraging results, with the model's accuracy soaring to an impressive 93%. The enhancement in performance is not limited to accuracy alone; other key metrics, detailed in the table below, also demonstrate significant improvement.

However, this uptick in performance came at a considerable computational cost. The fine-tuning process for the model spanned a substantial duration of approximately 9 to 10 hours. Upon reflection, we contend that the gain of 5% in accuracy may not justify such an extensive time investment. We believe that deploying a strategy of increased epochs over a shorter timeframe could potentially achieve similar, if not greater, enhancements in accuracy.

Below, you will find a detailed presentation of the evaluation results for our best-performing model, showcasing the outcomes of the five-fold cross-validation process.

Metric	Value
Accuracy	93
Precision	92
Recall	90
F1 Score	88

Confusion Matrix



Impact of Learning Rate Adjusment

The learning rate is a crucial hyperparameter in the fine-tuning of machine learning models, particularly deep learning architectures like BERT. It dictates the size of the steps the model takes during optimization. If the learning rate is set too high, the model may overshoot the optimal solution; if it's too low, the model may take too long to converge or get stuck in a local minimum. Adjusting the learning rate can have a significant impact on model performance. A well-tuned

learning rate can lead to faster convergence and potentially better generalization on unseen data. In the author identification task, experimenting with different learning rates could help find the sweet spot where the model learns patterns distinctive to each author without overfitting to the training data.

Effect of Omitting Stopwords

Stopwords in Persian, much like their English counterparts, are frequent words that usually have little unique contextual significance on their own—examples include "و" (and), "در" (in), "است" (is), among others. In a range of natural language processing endeavors, these stopwords are often filtered out to condense the dataset and to sharpen the model's focus on more meaningful words that carry greater informational weight. This practice is particularly nuanced in the realm of author identification.

The distinctive usage patterns of Persian stopwords could very well be integral to an author's stylistic fingerprint. Removing these words could inadvertently strip away subtle stylistic nuances, thus impairing the model's finesse in discerning one author from another. We investigated this by fine-tuning a multilingual BERT base model on Persian text data, initially without any preprocessing to remove stopwords. When we compared this to a fine-tuned model on data that had undergone preprocessing—specifically, the elimination of stopwords—the latter's performance was approximately 23 percent lower. This significant disparity suggests that, at least for this dataset and task, stopwords carry a critical stylistic value that aids in author differentiation. It underlines the fact that the removal of Persian stopwords could

potentially streamline the learning process; however, it also underscores the risk of losing crucial stylistic cues that contribute to the model's predictive accuracy.

Influence of Document Length

Document length can be a determining factor in the performance of text classification models. Shorter documents may not provide enough information for the model to accurately identify the author, while longer documents may introduce noise or irrelevant information. Furthermore, the attention mechanism in models like BERT has a maximum sequence length, and longer documents might need to be truncated, potentially losing valuable information. Balancing the document length to include enough stylistic detail for author identification without overwhelming the model or exceeding token limits is crucial. It's possible that there is an optimal document length that captures the necessary stylistic features while minimizing noise, thus enhancing the model's performance.

In summary, tuning the learning rate, deciding whether or not to omit stopwords, and determining the optimal document length are all important considerations that can significantly affect the outcome of an author identification model. Each of these factors should be carefully experimented with to evaluate their impact on the model's ability to learn and generalize from the data.

Traditional ML Approaches

Naïve Bayes

In our recent analysis, we employed the Naive Bayes (NB) classifier for author identification, continuing the methodology from a prior mini-project. Naive Bayes is known for its simplicity and efficiency, operating directly on raw text without the need for intricate embeddings, which is a significant advantage in terms of model complexity and computational demands. Remarkably, the model fitting process was expeditious, clocking in at merely 7 to 8 minutes. The NB model exhibited exceptional performance, achieving approximately 99% accuracy on our dataset.

This level of accuracy prompted a rigorous re-examination of the dataset to ensure that no inadvertent data leakage had occurred, such as overlapping test and training sets or any discernible hints of the author's identity within the documents. We can confidently assert that such issues were absent, solidifying the integrity of our results.

Despite the impressive performance of the Naive Bayes model, we speculate that the potential of BERT-based models could be even greater. BERT's deep learning architecture is designed to capture complex contextual relationships within text, which is likely to excel in tasks requiring nuanced understanding, such as distinguishing between different authors' writing styles. At the end you can see NB features down below:

Naive Bayes Advantages

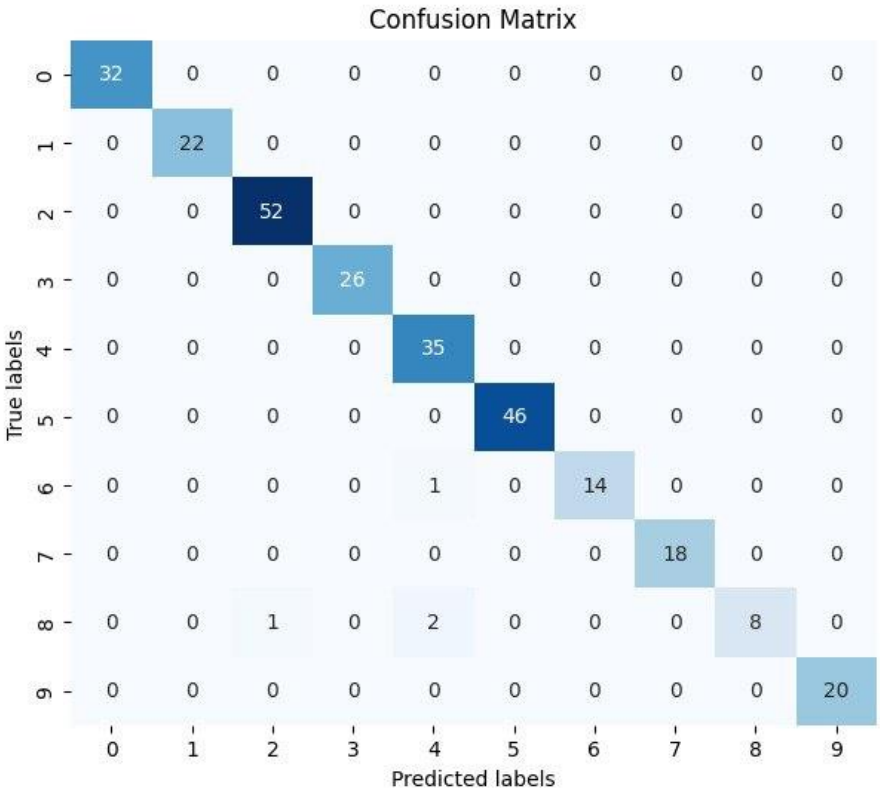
- **Speed**: Fast training times are advantageous for rapid model development and iteration.
- **Simplicity**: Easy to implement and understand, making it a good baseline model.
- **Efficiency**: Requires less computational power, making it suitable for constrained environments.

Naive Bayes Disadvantages

- **Assumption of Independence:** The model's assumption that features are independent can limit its performance on data where this condition is not met.
- **Feature Relationships:** Struggles to capture complex relationships in text due to its simplicity.

Metric	Value
Accuracy	99
Precision	99
Recall	99
F1 Score	98

NB Confusion Matrix



SVM

In our exploration of classification models for author identification, we incorporated Support Vector Machines (SVM) with a feature limit set to a maximum of 100, utilizing a linear kernel to parse the text data. The SVM model offered a notable advantage in terms of efficiency, it required only about 15 minutes to fit to our dataset, which is significantly faster compared to the more computationally intensive BERT model. After fitting, the SVM model demonstrated commendable efficacy, yielding an accuracy of 87%. This level of performance

positions SVM as a competitive alternative, especially when considering the balance between speed and accuracy.

When we consider the advantages and disadvantages of using SVM for such a task:

SVM Advantages

- **Efficiency**: SVMs are relatively fast to train on moderate-sized datasets, especially with linear kernels.
- **Effectiveness**: They are known for their effectiveness in high-dimensional spaces, even with a limited number of features.
- **Versatility**: The choice of kernel allows SVMs to adapt to different types of data distributions and relationships.

SVM Disadvantages

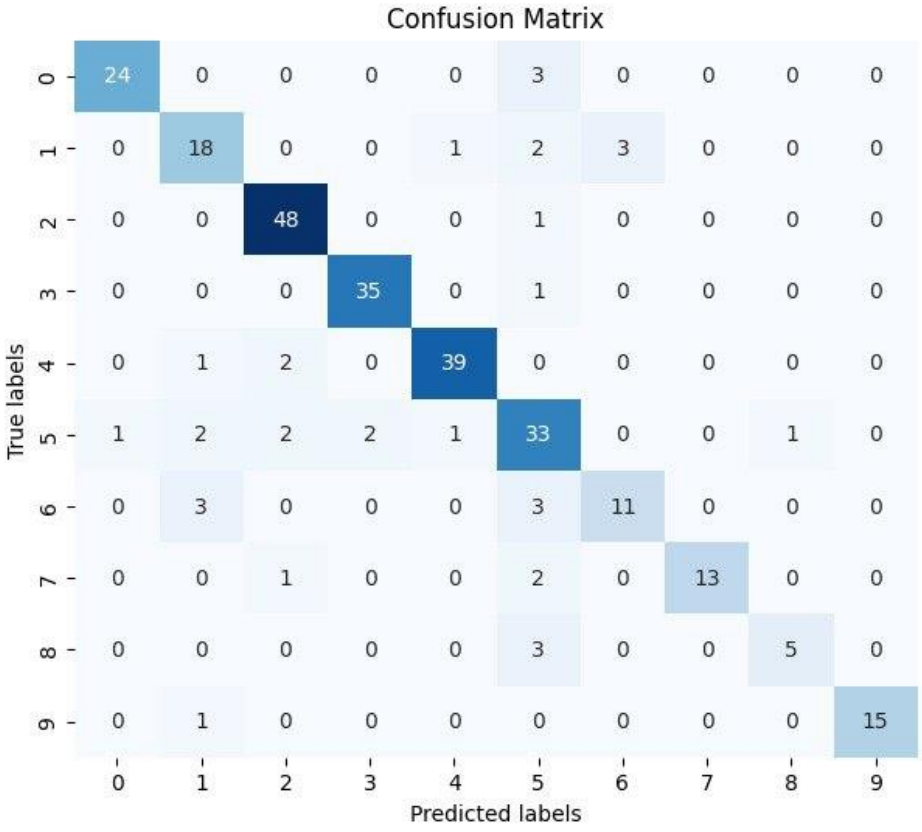
- **Scalability:** SVMs can become increasingly challenging to train efficiently as the size of the dataset grows, particularly with non-linear kernels.
- **Parameter Tuning:** Selecting the appropriate kernel and tuning the model parameters can be complex and requires a good understanding of the data.
- **Limited Interpretability:** Understanding how an SVM model makes its decisions can be less intuitive than some other models, such as decision trees.

Overall, while the SVM did not outperform the BERT (almost) model in terms of accuracy, its swifter training time and strong performance make it an attractive model for scenarios where computational

resources or time are limiting factors. However, for tasks demanding the highest accuracy where resources are abundant, BERT's superior contextual understanding may make it the preferred choice despite its higher computational demands.

Metric	Value
Accuracy	87
Precision	87
Recall	87
F1 Score	87

SVM Confusion Matrix



While both SVM and Naive Bayes (NB) have shown commendable performances in our author identification task, with SVM providing a good balance between speed and accuracy and NB impressing with its high accuracy and swift training times, BERT stands out as the superior choice for a few critical reasons. BERT's deep learning architecture is adept at capturing the nuances of language, providing a more sophisticated understanding of context and the subtle differences in authorial style. This is particularly important in author identification, where the way language is used can be as distinctive as a fingerprint. BERT's ability to process a wide range of contextual clues and its use of transfer learning—where knowledge from one domain is applied to another—allow it to achieve a level of precision that traditional models like SVM and NB may not reach. Although BERT requires more computational resources, the investment is justified by its ability to discern intricate patterns and produce results with higher confidence, making it an invaluable tool for complex NLP tasks where the depth of linguistic comprehension is paramount. At the end here are BERT features in compare to traditional ML approaches:

BERT Advantages

- **Contextual Understanding:** Capable of understanding complex word relationships and context, which is highly beneficial for NLP tasks.
- **Transfer Learning:** Pre-trained models can be fine-tuned on specific tasks, leveraging large datasets BERT was originally trained on.

BERT Disadvantages

- **Computational Cost:** Requires significant computational resources, especially for training from scratch or fine-tuning.
- **Complexity:** More complex to set up and fine-tune correctly, requiring a deeper understanding of the model's architecture and parameters.

Memorable Experiment

One of the most striking revelations from our series of experiments was the profound impact of BERT-based models in the field of natural language processing. While their performance is undeniably impressive, capturing the subtleties of language with remarkable accuracy, they come with a notable caveat—their requirement for considerable computational time to fine-tune. This aspect of BERT models underscores a significant trade-off in the NLP domain: the higher the model's capacity to understand and process language, the more substantial the time investment needed for fine-tuning. This relationship between performance and time investment is a critical consideration when deploying such advanced models in real-world applications.

Conclusion

In this report, we have provided an overview of our investigative journey into the application of machine learning models for the purpose of author identification. Our team successfully curated a dataset tailored to the intricacies of Persian text and embarked on fine-tuning BERT models, paying careful attention to the roles of learning rates and document lengths. We examined and contrasted the performances of traditional algorithms such as Naive Bayes and SVM, appreciating their rapid execution and straightforward implementation, against the backdrop of BERT's deep learning prowess. Despite the heavier computational demands of BERT models, our team concluded that their unparalleled capacity for contextual analysis offers a significant advantage in capturing the essence of linguistic styles, solidifying their status as a leading choice for complex natural language processing challenges.