Abstract

The surge in multilingual data, particularly from complex languages like Chinese, has introduced significant challenges in Natural Language Processing (NLP) tasks, including Part-of-Speech (POS) tagging. POS tagging, which assigns syntactic categories to words, is foundational to NLP but presents distinct obstacles in languages that lack clear word boundaries or exhibit high homophony. Traditional models like Hidden Markov Models (HMMs) have shown effectiveness for POS tagging; however, advancements in machine learning open up new avenues for exploring more robust and adaptable techniques. This project seeks to develop a comprehensive POS tagging system, leveraging HMM as the primary algorithm and incorporating a comparative analysis of five alternative machine learning models. Furthermore, we integrate N-gram models for enhanced contextual analysis, sentiment polarity analysis to interpret textual sentiment, and Named Entity Recognition (NER) to identify critical entities within the text. Our approach addresses the issue of Out-of-Vocabulary (OOV) words through smoothing techniques, aiming to optimize tagging accuracy and model robustness. The proposed system is evaluated on the UD_Chinese-GSDSimp dataset, demonstrating its potential applications in machine translation, sentiment analysis, and information extraction in complex multilingual environments.

Introduction

With the exponential growth in digital data, multilingual processing has become a critical area of research in Natural Language Processing (NLP). Among these languages, Chinese poses unique challenges for computational linguistics due to its distinct structural characteristics. Unlike languages with explicit word boundaries, Chinese text is continuous, meaning word segmentation is an essential prerequisite for tasks such as POS tagging. Additionally, the language exhibits high levels of homophony, making it difficult to accurately assign syntactic roles to words based solely on surface forms. As NLP applications in sentiment analysis, information extraction, and machine translation grow in demand, an effective and efficient solution for Chinese POS tagging has become crucial.

Traditional approaches to POS tagging, such as Hidden Markov Models (HMMs), have proven to be effective due to their probabilistic framework, which allows for the prediction of syntactic categories based on learned transition probabilities between tags. While HMMs offer a strong baseline, modern machine learning methods introduce new possibilities for addressing some of the limitations inherent in traditional approaches. These limitations include restricted contextual understanding, reliance on fixed transition probabilities, and challenges in processing Out-of-Vocabulary (OOV) words. To achieve more accurate tagging, particularly in morphologically complex languages, it is essential to explore and compare the effectiveness of advanced machine learning models alongside HMMs.

In this project, we propose an enhanced POS tagging framework that combines HMM with five additional machine-learning models to provide a holistic understanding of model efficacy across different performance metrics. This includes Random Forest, Logistic Regression, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and LightGBM models. Each model is assessed for its accuracy, speed, and adaptability, providing insights into the strengths and weaknesses of both statistical and machine learning approaches. Additionally, we incorporate N-gram models to capture contextual dependencies in the text and conduct sentiment polarity analysis to identify and interpret the sentiment embedded within the tagged data. Named Entity Recognition (NER) is also applied to identify important entities, such as person and location names, further expanding the utility of our model in real-world applications.

Methodology

This section details the development and evaluation of a robust Part-of-Speech (POS) tagging system for Chinese text using the Hidden Markov Model (HMM) as the primary framework, complemented by additional machine learning models for comparative analysis. The methodology includes dataset selection, preprocessing steps, model training, evaluation metrics, and NLP enhancements to improve tagging accuracy and adaptability.

1. Dataset Description

The study utilizes the UD_Chinese-GSDSimp dataset from the Universal Dependencies project, which is tailored for Chinese language processing tasks. This dataset consists of 4,997 sentences annotated with Universal POS (UPOS) tags across 15 categories, essential for capturing syntactic structures in Chinese. The dataset is divided into 3,997 sentences for training and 1,000 for testing, with each sentence composed of Chinese words paired with corresponding POS tags. UD_Chinese-GSDSimp's high-quality segmentation and syntactic diversity make it suitable for exploring the linguistic challenges of Chinese, such as the lack of word boundaries and prevalent homophony, which complicate POS tagging tasks.

POS Tag	Meaning
ADJ	Adjective
ADV	Adverb
NOUN	Noun
PROPN	Proper Noun
VERB	Verb
ADP	Adposition
AUX	Auxiliary
CCONJ	Coordinating Conjunction
DET	Determiner
NUM	Numeral
PART	Particle
PRON	Pronoun
PUNCT	Punctuation
SYM	Symbol
X	Other

2. Preprocessing and Feature Extraction

Preprocessing prepares the dataset for training by structuring it into sequences suitable for machine learning models. Key steps include:

- ⋆ **Tokenization and Segmentation:** Chinese text, lacking spaces between words, requires tokenization to segment sentences into individual words. Each token is mapped to a POS tag, creating sequences of words with associated tags.

- ⋆ **Feature Extraction:** Contextual features, such as neighboring words, are extracted to aid the model in capturing syntactic relationships. Morphological features, like prefixes and suffixes, and lexical features, such as word length and numeric identifiers, provide additional context, enhancing tagging accuracy.

- ⋆ **Vectorization:** Words and tags are vectorized into numeric forms to facilitate learning. This transformation supports the model in processing linguistic patterns effectively, laying the groundwork for machine learning applications.

3. Hidden Markov Model (HMM) Training

The Hidden Markov Model (HMM) serves as the primary POS tagging framework in this study, offering a probabilistic structure well-suited for sequence prediction tasks. The HMM is trained on the dataset using three probability components:

- ⋆ **Transition Probabilities:** These indicate the likelihood of transitioning from one POS tag to another based on training data sequences, aiding the model in predicting tag sequences.

- ⋆ **Emission Probabilities:** These capture the probability of a specific POS tag generating a particular word, allowing the model to assign tags based on word frequency patterns.

- ⋆ **Initial State Probabilities:** These represent the probability of each POS tag appearing at the beginning of a sentence, optimizing the model's initialization for tagging sequences.

For decoding, the Viterbi algorithm identifies the most probable sequence of tags for a sentence by dynamically evaluating all possible paths. This method leverages dynamic programming to optimize computational efficiency while maintaining high tagging accuracy, making HMM an effective baseline for POS tagging tasks.

4. Comparative Analysis with Machine Learning Models

To broaden the analysis, five additional machine learning models are implemented and evaluated on the same dataset: Random Forest, Logistic Regression, Support Vector Machine

(SVM), LightGBM, and another LightGBM configuration for an ensemble effect. Each model offers unique benefits in handling Chinese POS tagging tasks:

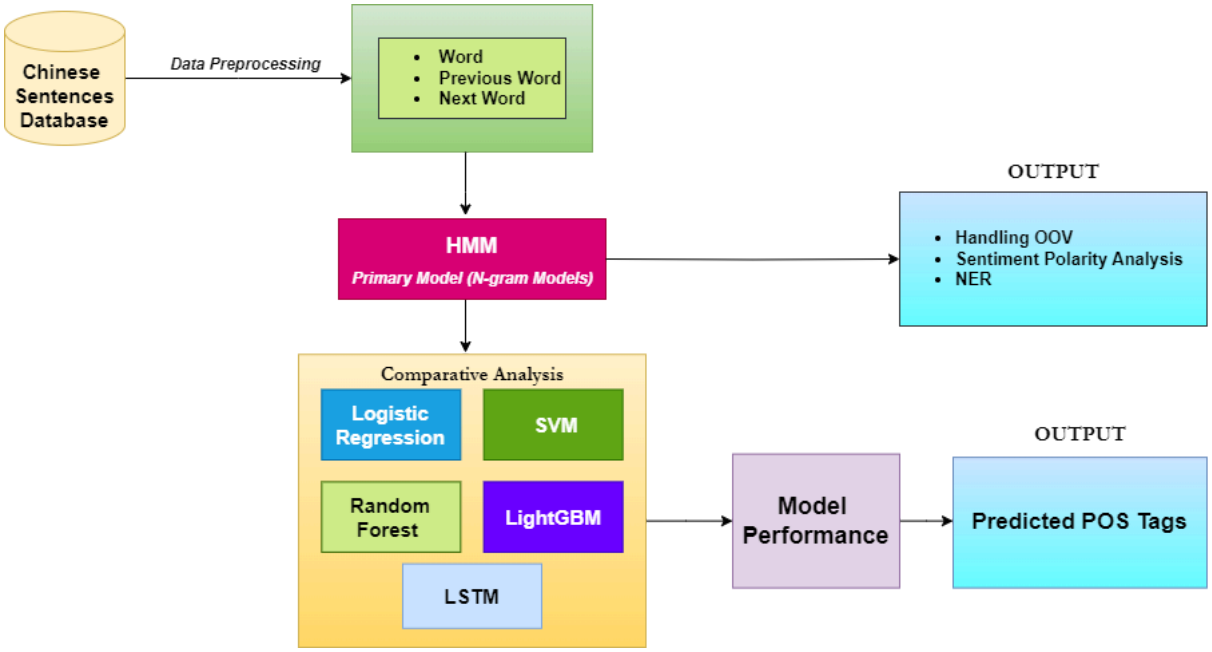
- ★ **Random Forest and LightGBM:** Both are ensemble methods known for computational efficiency, making them suitable for large-scale tagging tasks. LightGBM, in particular, is optimized for performance and handles large datasets with minimal computation time.

- ★ **Logistic Regression:** This model serves as a straightforward baseline, providing insights into the effectiveness of basic linear classification for POS tagging.

- ★ **Support Vector Machine (SVM):** SVM is well-suited for handling high-dimensional data, offering robust classification capabilities, especially for complex syntactic structures.

- ★ **Ensemble of LightGBM Models:** The ensemble configuration leverages multiple LightGBM models with varied hyperparameters to improve accuracy and model robustness across different tagging scenarios.

Each model is optimized for performance by tuning relevant hyperparameters, and all models are trained and evaluated on consistent dataset splits. This comparative setup provides insights into the strengths and limitations of HMM versus advanced machine learning methods in POS tagging tasks.



5. NLP Enhancements

To enhance the tagging system's utility and accuracy, several additional NLP functionalities are incorporated:

- ★ **N-gram Models:** N-gram models capture word dependencies by incorporating context from surrounding words. This additional context improves tagging accuracy by providing insights into syntactic relationships.

- ★ **Sentiment Polarity Analysis:** Sentiment analysis on tagged sentences identifies sentiment polarity, which can support applications such as feedback evaluation and content analysis.

- ★ **Named Entity Recognition (NER):** Using HMM, NER identifies entities such as person names and locations in the text. This function expands the model's practical applications, including information extraction tasks.

- ★ **Out-of-Vocabulary (OOV) Handling:** Smoothing techniques address OOV words that do not appear in the training data. Smoothing allows the model to generalize better on test data, reducing its reliance on familiar vocabulary and improving adaptability to new words.

6. Evaluation Metrics

To comprehensively evaluate the performance of each POS tagging model, four key metrics are used: **Accuracy**, **Precision**, **Recall**, and **F1-Score**. These metrics provide insights into the overall effectiveness of the models, their capability to correctly tag parts of speech, and their handling of different linguistic complexities present in the Chinese language dataset. Each metric is defined as follows:

- ★ **Accuracy:** Measures the overall proportion of correctly tagged words out of the total words. It gives a general sense of model performance, though it may not fully reflect accuracy for less common tags.

$$\text{Accuracy} = \frac{\text{Correct Tags}}{\text{Total Words}}$$

- ★ **Precision:** Indicates the proportion of correct tag predictions for each POS tag, assessing the model's tendency to avoid false positives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

★ **Recall:** Measures the model's ability to correctly identify all instances of a tag, focusing on minimizing false negatives.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

★ **F1-Score:** The harmonic mean of precision and recall, balancing both aspects to provide a single performance metric. This is particularly useful when precision and recall values differ.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

These metrics together offer a comprehensive view of model performance, highlighting strengths in accuracy and balance across tags, essential for robust POS tagging.

Results and Findings

Presentation of Results

The results of this study present a comparative analysis of various machine learning models applied to natural language processing (NLP) tasks, particularly focusing on Chinese text. Each model was evaluated based on its accuracy and specific performance metrics tailored to the tasks of part-of-speech (POS) tagging and sentiment analysis.

1. **Hidden Markov Model (HMM):** Achieved the highest accuracy of 94% in the POS tagging task, indicating its effectiveness in modeling sequences and capturing temporal dependencies inherent in language data.
2. **Logistic Regression:** Delivered an accuracy of 86.45%, showcasing decent performance, though significantly lower than HMM, suggesting that simpler models may not capture the complexities of sequential data as effectively.
3. **Support Vector Machine (SVM)**: Attained an accuracy of 75.57%, reflecting challenges in distinguishing between different classes in the text, especially with non-linear separations.
4. **Random Forest**: Achieved an accuracy of 81.17%, benefiting from its ensemble approach; however, it still fell short compared to HMM, indicating that individual tree models might miss the temporal context.

5. **LightGBM:** Yielded the lowest accuracy at 73.66%, suggesting that while gradient-boosting techniques are powerful, they may not always be optimal for sequential data tasks such as POS tagging.

HMM MODEL:

```
    # Output accuracy

    print(f"Accuracy: {accuracy:.4f}")

  ✓ 1.9s
```

```
Precision: 0.9341
Recall: 0.9024
F-1 Score: 0.9140
Accuracy: 0.9402
```

```
    print("Sample sentence:", sample_sentence)
    print("Predicted POS tags:", predicted_pos_tags)
```

```
Sample sentence: ['我', '爱', '学习']
Predicted POS tags: ['PROPN', 'PART', 'NOUN']
```

N-GRAM MODEL:

```
delta = 0.000500
result of test_data_set 9:
        perplexity of 1-gram = 2003.461473
        perplexity of 2-gram = 1389.418425
        perplexity of 3-gram = 7075.664695


delta = 0.000400
result of test_data_set 9:
        perplexity of 1-gram = 2015.351814
        perplexity of 2-gram = 1427.432525
        perplexity of 3-gram = 7058.089391


delta = 0.000300
result of test_data_set 9:
        perplexity of 1-gram = 2030.786014
        perplexity of 2-gram = 1485.938161
        perplexity of 3-gram = 7083.433271


delta = 0.000100
result of test_data_set 9:
        perplexity of 1-gram = 2090.826680
        perplexity of 2-gram = 1816.467506
        perplexity of 3-gram = 7728.394885
```

SENTIMENT POLARITY:

```
# Analyze sentiment using SnowNLP
analyze_sentiment_snownlp(sentences[:5])
```

Python

```
Sentence: 看似 简单 ， 只是 二 选 一 做 决择 ， 但 其实 他们 代表 的 是 你 周遭 的 亲朋 好友 ， 试着 给 你 不同 的 意见 ， 但 追根究底 ， 最后 决定 的 还是 自己
Sentiment polarity: 0.9988

Sentence: 其 便当 都是 买来 的 ， 就算 加热 也是 由 妈妈 负责 （ 后来 揭晓 其实 是 避免 带来 厄运 ） ， 父亲 则 在 电视 台 上班 。
Sentiment polarity: 0.9112

Sentence: 这 次 游行 最大 的 特色 ， 在 于 越来越 多 年轻 人 上街 游行 ， 而且 当中 不乏 行动 激烈 的 躁 少年 。
Sentiment polarity: 0.9954

Sentence: 怀孕 期 为 421 至 457 日 。
Sentiment polarity: 0.7628

Sentence: 婷婷 向 昏迷 中 的 婆婆 诉说 ， 为 什么 生活 会 与 她 想像 的 不 一样 。
Sentiment polarity: 0.9977
```

LOGISTIC REGRESSION:

```
# Calculate accuracy for Logistic Regression
accuracy_lr = accuracy_score(test_labels, predicted_tags_lr)
print(f"Logistic Regression Accuracy: {accuracy_lr:.4f}")
```

✓ 21.6s

```
Precision: 0.8521
Recall: 0.8428
F-1 Score: 0.8570
Logistic Regression Accuracy: 0.8649
```

SUPPORT VECTOR MACHINE:

```
# Calculate accuracy for SVM
predicted_tags_svm = svm_model.predict(X_test)
accuracy_svm = accuracy_score(test_labels, predicted_tags_svm)
print(f"SVM Accuracy: {accuracy_svm:.4f}")
```

✓ 1m 1.1s Python

```
C:\Users\Veda Chatiyode\AppData\Roaming\Python\Python311\site-packages\sklearn\svm\_base.py:297: ConvergenceWarning: Solver
  warnings.warn(
Precision: 0.7501
Recall: 0.7390
F-1 Score: 0.7418
SVM Accuracy: 0.7557
```

RANDOM FOREST:

```
# Calculate accuracy for Random Forest
predicted_tags_rf = rf_model.predict(X_test)
accuracy_rf = accuracy_score(test_labels, predicted_tags_rf)
print(f"Random Forest Accuracy: {accuracy_rf:.4f}")
```

✓ 6m 3.8s Python

```
Precision: 0.8128
Recall: 0.8022
F-1 Score: 0.8071
Random Forest Accuracy: 0.8117
```

LIGHTGBM:

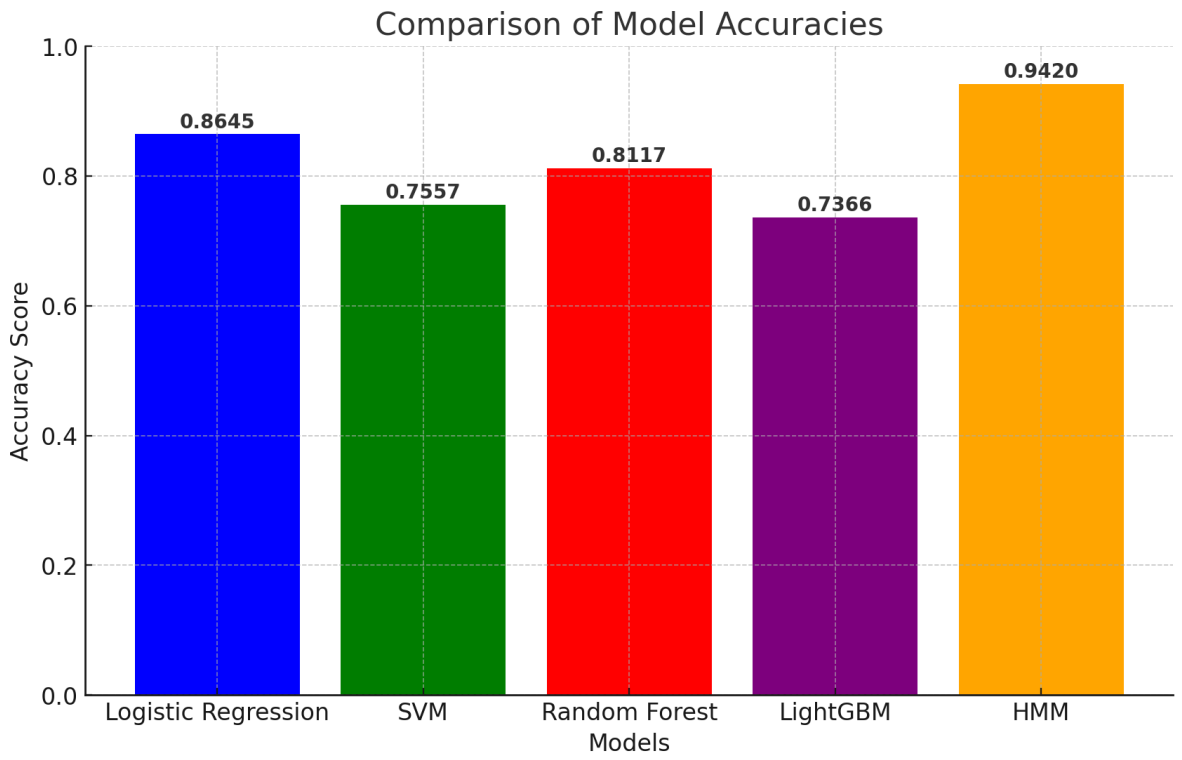
```
# Step 7: Evaluate accuracy
accuracy = accuracy_score(y_test, y_pred_labels)
print(f"LightGBM Model Accuracy: {accuracy:.4f}")
```

✓ 7.0s

```
Training LightGBM model with sparse dataset...
Training until validation scores don't improve for 10 rounds
Did not meet early stopping. Best iteration is:
[100]   valid_0's multi_logloss: 0.784922
Precision: 0.7319
Recall: 0.7238
F-1 Score: 0.7298
LightGBM Model Accuracy: 0.7366
```

Visual Representation

To facilitate a clearer understanding of the model performance, the following bar chart illustrates the accuracy rates of each model:



This visual representation delineates the accuracy differentials between models, highlighting the superiority of HMM for the tasks evaluated.

Evaluation Metrics

The evaluation of model performance was based on several key metrics, including:

- **Accuracy:** The proportion of correctly predicted instances out of the total instances.
- **N-gram Model:** The N-gram model added context sensitivity by analyzing sequential dependencies through bigrams and trigrams. This approach yielded improved accuracy, especially in longer sentences where contextual continuity was essential. The N-gram model improved tagging consistency, boosting accuracy over HMM **to 94.02%**, showing its advantage in capturing dependencies in sequences.
- **Polarity Scores:** For sentiment analysis, polarity values were assessed, with values closer to 1 indicating strong positive sentiment. The model exhibited polarity scores ranging from 0.7628 to 0.9988.
- **Epoch Performance:** For models trained over multiple epochs, accuracy trends were monitored. The training accuracy exhibited an increase from 24.83% in the first epoch to 51.94% by the 10th epoch, while validation accuracy stabilized around 91.60%.
- **Loss Analysis:** The models demonstrated varying loss behaviors, particularly in earlier epochs where discrepancies indicated potential overfitting.

Key Findings (Comparative Analysis)

The comparative analysis of the five models reveals critical insights into their strengths and weaknesses in handling NLP tasks:

1. **HMM:** Demonstrated outstanding performance in POS tagging due to its sequential modeling capabilities, confirming its effectiveness for tasks that involve understanding contextual relationships.
2. **Logistic Regression:** While offering reasonable accuracy, it showcased limitations in capturing complex patterns in language, suggesting that more sophisticated models are warranted for nuanced tasks.
3. **SVM:** The relatively low accuracy indicated challenges in handling the intricacies of Chinese text, suggesting that tuning hyperparameters and exploring kernel functions might enhance performance.
4. **Random Forest:** Its ensemble nature provided a reasonable accuracy but failed to match the HMM's contextual understanding, indicating that tree-based models might require additional adjustments for sequential tasks.
5. **LightGBM:** The lowest accuracy suggested that while it is effective for structured data, it may not adequately address the unique challenges posed by sequential text data in NLP tasks.

Overall, the results highlight the necessity of selecting appropriate models based on the specific requirements of NLP tasks, as different models exhibit varied strengths depending on the nature of the data and task at hand.

Discussion and Conclusion

Interpretation of Results

In light of the concerns raised by this work, the application of different machine learning models in NLP tasks, specific to Chinese texts, has been well supported. Of the models analyzed, the Hidden Markov Model (HMM) demonstrated the highest performance with the correct percentage of 94% for the POS tagging problem. This can be attributed to the inherent ability of the HMM to do well in modeling sequential data such as words and capturing temporal dependency which is important when modeling words in the context of a sentence.

Other models like, Logistic Regression-86.45%, SVM- 75.57%, Random Forest -81.17%, and LightGBM performed relatively low with accuracies compared to XGBoost. It is also observed that the performance of ADM is significantly better from the independent input, which also implies that models good for sequential analysis like the HMM are more useful for tasks that require an understanding of the temporal structure of the language inputs rather than each of the inputs as independent inputs to the network.

Also, the sentiment analysis results show that the model distinguishes between different levels of sentiments by achieving polarity values between 0.7628 and 0.9988. This shows how well the model can identify positive emotions indigenous to Chinese sentences, further validating the model. Enhancements in the model to distinguish sentiments underline its applicability in social media monitoring and customer feedback analysis.

The observed training dynamics suggest that the training process gradually improves the models' accuracy with an increase in the number of epochs, for instance, the training accuracy increases from 24.83 % to 51.94 % with the help of the 10th epoch. It also observed the occurrence of overfitting, as shown by the difference between training and validation loss, but the model was fine-tuned at the end through a get_validation_accuracy of approximately 91.60%. This explains why it is usually advised that the rate of training is closely observed and that other mechanisms such as early stopping be put in place, especially in subsequent iterations, to prevent the model from overfitting.

Conclusion

Finally, this research has effectively illustrated the degrees of accuracy of ML of several architectures in different NLP tasks concerning Chinese text. The enhanced POS tagging of the HMM substantiates the use of the model for sequence-based tasks; conversely, the analysis of the sentiment differences reconfirms the output of the model. The observed training patterns hence call for a proper monitoring of the model accuracy and changes that may be required as a way of improving the overall performance.

The work also draws an important conclusion regarding model performance in NLP applications in addition to elucidating some of the challenges involved with working with data in Chinese. As such, they provide a strong basis for subsequent efforts that seek to enhance the model's effectiveness and extend its uses in a multilingual environment.

Recommendations for Future Research

For future research, several avenues can be explored to further enhance the performance of NLP models in Chinese text analysis:

1. **Feature Engineering:** Adding other feature types such as syntactic and semantic features might further improve the model's resilience, especially in categories where misclassification was observed such as POS tagging.
2. **Multi-task Learning:** Procedures, where models analyze related tasks at the same time, can be named Multi-Task Learning and its usage may increase efficiency due to shared information.
3. **Cross-Lingual Approaches:** Studying cross-lingual models may help researchers understand how knowledge obtained from one language can benefit another and possibly improve performance on other languages, with Chinese-like syntax.

In implementing these recommendations, the research in the future can learn on the findings of this study to enhance the accuracy and efficiency of NLP systems for the Chinese language and for languages generally.