Research on Relation Extraction with SVM and BERT

Zhuoxuan He; Kunwei Song; Xingjian Yuan
Department of Computer Science
University of Manchester

1 Introduction

Relation Extraction (RE) is a fundamental task in Natural Language Processing (NLP) that focuses on identifying semantic relationships between entities within a given sentence. It is a key component of various NLP applications, including information extraction, knowledge graph construction, and question answering systems (Bassignana & Plank, 2022). Over the years, research on relation extraction has evolved from early statistical and feature-based machine learning methods such as support vector machines (SVMs) to modern deep learning methods driven by pre-trained language models such as BERT, which utilize contextual representations to improve performance ((Devlin et al., 2019). In addition, some studies have explored hybrid techniques that combine traditional and deep learning models to leverage the strengths of each.

To advance RE research, various benchmark datasets have been developed to facilitate model evaluation and comparison. In this study, we adopt SemEval 2010 Task 8 as our experimental dataset (Hendrickx et al., 2010). This dataset is specifically designed for multi-way classification of semantic relations between entity pairs and contains manually annotated sentences labelled with 19 relation types, including additional categories for cases where no meaningful relations exist. It consists of 8,000 training samples and 2,717 test samples, making it a reliable dataset for evaluating RE models.

In this study, we developed and improved two different RE models based on the SemEval 2010 Task 8 dataset. The first is a hybrid SVM-based model that integrates TF-IDF for text enhancement, BiLSTM with Attention for feature extraction (Zhou et al., 2016), and GloVe pre-trained word embeddings to enhance the model's understanding of semantic relations. The second is an improved BERT-based model, which employs contrastive learning (Gao et al., 2021) to make samples of the same class more tightly clustered and incorporates external knowledge from WordNet and Wikidata (Peters et al., 2019; Yao et al., 2019) to improve generalization.

To evaluate the effectiveness of the proposed methods, we use Precision, Recall, and F1-score as performance metrics. Precision measures the accuracy of classification, while Recall reflects the model's ability to capture correct relations. F1-score balances both metrics, providing a more comprehensive assessment of overall performance. Experimental results demonstrate that our approach significantly outperforms the baseline across multiple relation categories. Moreover, our analysis highlights the strengths and limitations of traditional machine learning methods (SVM) and deep learning approaches (BERT) in the RE task, providing a clearer understanding of their practical advantages and challenges.

The remainder of this paper is structured as follows: Section 2 reviews related work, Section 3 introduces the methodology, Section 4 presents results and discussion, and Section 5 concludes the study.

2 Review of Related Work

Relation Extraction (RE) has evolved from symbolic methods relying on hand-crafted rules to modern deep learning approaches. Early work employed rule-based systems and bootstrapping techniques, but these suffered from limited adaptability (Bassignana & Plank, 2022). Traditional machine learning models, such as Support Vector Machines

(SVMs), introduced feature engineering (Zhou et al., 2005; Hong, 2005). Kernel methods further leveraged syntactic structures for improved performance (Zelenko et al., 2003). However, these approaches often struggled with complex sentence structures and long-range dependencies.

Deep learning methods addressed some of these challenges by automating feature extraction with RNNs, CNNs, and LSTMs (Zhang et al., 2015). BiLSTM-based models improved sequential encoding, and attention mechanisms helped capture long-distance information (Ni et al., 2021). Yet, these architectures sometimes lacked sufficient global context.

Transformer-based models, especially BERT, substantially advanced RE via self-attention and contextual embeddings (Wu & He, 2019; Thillaisundaram & Togia, 2019). For instance, R-BERT incorporated explicit entity information into BERT, demonstrating the importance of entity-level features in improving RE performance (Wu & He, 2019). Nonetheless, BERT-based models can struggle with semantically similar relations and often neglect external knowledge (Peters et al., 2019; Yao et al., 2019).

In our work, we address these gaps by combining SVM with BiLSTM-based feature extraction, and by augmenting BERT with external knowledge and contrastive learning. These refinements aim to improve semantic understanding, reduce confusion among closely related classes, and enhance overall RE performance.

3 Methodology

3.1 SVM-Based Method

SVM is effective in high-dimensional spaces and is widely used in RE tasks, attributed to its solid theoretical foundation (Cortes & Vapnik, 1995). However, to improve its feature representation, we integrate BiLSTM with Attention for encoding text sequences before classification. This approach allows the model to capture long-range dependencies that conventional feature-based methods often fail to model effectively.

The implementation follows a structured pipeline. First, we load the SemEval 2010 Task 8 dataset, extracting sentences as features and relations as labels. The initial SVM implementation uses CountVectorizer to transform the text into word frequency vectors before training the classifier. Earlier studies have combined SVM with kernel methods for relation extraction tasks, achieving promising results (Zelenko et al., 2003). However, this method was later improved by replacing CountVectorizer with TF-IDF vectorization, which provides more informative word representations by reducing the influence of frequently occurring words.

The second improvement introduces BiLSTM with Attention for feature extraction. Text sequences are padded to a uniform length, then processed through a model consisting of an input embedding layer, a BiLSTM layer for contextual encoding, an attention mechanism to emphasize key features, and a pooling layer for dimensionality reduction. The resulting embeddings are then passed to the SVM classifier.

Further enhancements include class weighting and GloVe embeddings to handle class imbalance and improve word representations. Since certain relation types appear less frequently in the dataset, class weighting adjusts the loss function to assign higher importance to underrepresented classes. Additionally, GloVe embeddings provide pre-trained word vectors, enriching the semantic representation of text.

3.2 BERT-Based Method

BERT is selected due to its ability to model contextual dependencies using self-attention mechanisms. However, traditional fine-tuning with a classification head struggle to distinguish semantically similar relations. To address this, we incorporate contrastive learning to enhance class separability and external knowledge integration to improve semantic understanding.

Unlike the SVM model, BERT requires a different preprocessing approach. The SemEval 2010 Task 8 dataset already contains <e1> and <e2> entity markers, but these need to be converted into a format compatible with BERT. We explicitly tag entities using WordPiece tokenization, ensuring the model can effectively process relational information.

To further enhance the dataset, we integrate WordNet and Wikidata as external knowledge sources. WordNet definitions are retrieved and appended to the sentence, enriching its semantic context (Joachims, 1998). Additionally, Wikidata relations are extracted via SPARQL queries, providing structured relational information. We adopted a method similar to Li and Wu (2020), directly incorporating WordNet synsets and hypernym information into the model representation. The final augmented sentence includes entity definitions and knowledge graph relations, improving the model's ability to differentiate relation types.

Example of transformed input:

"The <e1>company</e1> acquired the <e2>startup</e2>. [SEP] company: A business organization [KG: owns: subsidiary]. [SEP] startup: A newly founded business entity [KG: funded by: investors]."

The processed text is tokenized using BERT's WordPiece tokenizer and input into the BERT encoder. To further refine relation classification, we employ Supervised Contrastive Learning (SCL), which encourages representations of the same relation type to be closer while pushing apart different relation types (van den Oord et al., 2018). Instead of solely relying on cross-entropy loss, contrastive loss is introduced to reshape the embedding space, making relations more distinguishable. To optimize training, we use a linear learning rate scheduler with AdamW optimizer. The CLS token representation is used for final classification, leveraging BERT's pre-trained knowledge to improve RE performance.

3.3 Evaluation Metrics and Experimental Setup

To evaluate model performance, we use Precision, Recall, and F1-score, focusing on the accuracy of relation classification. The F1-score, as the harmonic mean of Precision and Recall, provides a single metric that balances both (Sokolova & Lapalme, 2009). Training and evaluation are conducted using Google Colab, leveraging GPU acceleration for faster computation. The models are implemented in PyTorch, with batch size and hyperparameters adjusted based on validation results.

3.4 Summary

This study presents a comparative analysis of SVM enhanced with BiLSTM, and BERT enhanced with contrastive learning and knowledge augmentation. The SVM-based model is well-suited for low-resource settings, where structured feature extraction plays a crucial role. Meanwhile, the BERT model demonstrates superior performance in capturing complex relational contexts. By integrating external knowledge from WordNet and Wikidata, both approaches achieve improved semantic understanding, leading to enhanced RE performance.

4 Results and Discussion

This section presents the experimental results of the two relation extraction approaches studied in this work: SVM-based and BERT-based models. We analyze the performance of both methods, highlighting the impact of TF-IDF vectorization, BiLSTM + Attention feature extraction, and external knowledge augmentation for SVM, as well as knowledge integration, contrastive learning, and learning rate scheduling for BERT. The models are evaluated using accuracy, precision, recall, and F1-score.

As shown in Figure A.1, The baseline SVM model achieved an accuracy of 53.77% and an F1-score of 46.06%, indicating challenges in relation classification. Its reliance on CountVectorizer-based frequency representations resulted in poor recall (42.77%), particularly for low-resource relation types such as Entity-Destination and Content-Container. The introduction of TF-IDF vectorization improved text representation, increasing accuracy, but long-distance dependencies remained an issue.

Further enhancement with BiLSTM + Attention allowed the model to capture sequential dependencies, leading to a performance boost (70.30% accuracy, 66.80% F1-score; Figure A.2). The most significant improvement came from external knowledge augmentation using GloVe embeddings, which enriched semantic representations. Despite these advancements, ambiguity in certain relations persisted, and the model's performance was still inferior to deep learning-based approaches.

The baseline BERT model significantly outperformed SVM, achieving 76.11% accuracy and 71.24% F1-score, highlighting the advantage of self-attention and contextual word representations. However, it struggled with relations requiring external knowledge and distinguishing semantically similar relations. To address these limitations, we integrated WordNet for entity-level enhancement and Wikidata for relation-level knowledge augmentation. WordNet provided entity definitions, improving entity disambiguation, while Wikidata offered structured knowledge, enhancing relation differentiation.

Supervised contrastive learning was introduced to refine the representation space, ensuring that similar relation types clustered together while dissimilar ones were pushed apart. This was particularly effective for semantically overlapping relations, reducing misclassification between Cause-Effect and Instrument-Agency. Learning rate scheduling further stabilized training, preventing overfitting and improving generalization.

The final BERT model achieved an accuracy of 85.24% and an F1-score of 81.01%, a significant improvement over the baseline BERT and SVM model. The macro-average recall increased from 72.35% to 82.05%, confirming that external knowledge and contrastive learning improved model robustness. For a detailed visualization of these results, please see Figure B.1 and B.2

To compare traditional and deep learning-based approaches, the best SVM model (70.30% accuracy, 66.80% F1-score) was contrasted with the final BERT model (85.24% accuracy, 81.01% F1-score). The SVM model struggled with contextual relationships and required manual feature extraction, whereas BERT automatically learned representations and benefited more from external knowledge augmentation. These results confirm the superiority of deep learning-based models for relation extraction, particularly in handling complex semantic relationships and context-dependent relations.

This study presents a comparative analysis of SVM enhanced with BiLSTM, and BERT enhanced with contrastive learning and knowledge augmentation. The SVM-based model is well-suited for low-resource settings, where structured feature extraction plays a crucial role. Meanwhile, the BERT model demonstrates superior performance in capturing complex relational contexts. By integrating external knowledge from WordNet and Wikidata, both approaches achieve improved semantic understanding, leading to enhanced RE performance.

5 Conclusion

This study demonstrates that relation extraction performance can be significantly enhanced through external knowledge integration, contrastive learning, and optimized training strategies. The SVM model, improved with TF-IDF, BiLSTM + Attention, and external knowledge embeddings, showed notable gains but remained limited by its reliance on manual feature extraction. In contrast, the BERT model, enhanced with WordNet, Wikidata, contrastive learning, and learning rate optimization, achieved 85.24% accuracy and 81.01% F1-score, outperforming both the baseline BERT and the best SVM model.

Despite improvements, challenges remain. Low-resource relation types still exhibit lower recall, and structured knowledge sources do not always provide sufficient coverage. Future work could explore dynamic knowledge retrieval for real-time knowledge integration and more advanced pre-trained models like DeBERTa or T5 to enhance contextual understanding.

These results confirm that transformer-based models, when optimized with knowledge augmentation and improved training strategies, outperform traditional machine learning methods in relation extraction. Future research should focus on scalable deep learning architectures and adaptive knowledge integration to further improve model robustness and generalization.

References

Bassignana, E., & Plank, B. (2022). What Do You Mean by Relation Extraction? A Survey on Datasets and Study on Scientific Relation Classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop* (pp. 67-83). Association for Computational Linguistics.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). 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* (pp. 4171–4186). Association for Computational Linguistics.

Hong, G. (2005). Relation Extraction Using Support Vector Machine. In *Proceedings of the Second International Joint Conference on Natural Language Processing: Full Papers.* Association for Computational Linguistics.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Oepen, S., Pado, S., & Yuret, D. (2010). SemEval-2010 Task 8: Multi-way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (pp. 33–38). Association for Computational Linguistics.

Zhou, J., & Xu, W. (2015). End-to-End Learning of Semantic Role Labeling Using Recurrent Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics.

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (pp. 6894–6910). Association for Computational Linguistics.

Peters, M. E., Neumann, M., Logan, R., Schwartz, R., Joshi, V., Singh, S., & Smith, N. A. (2019). Knowledge Enhanced Contextual Word Representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (pp. 43–54). Association for Computational Linguistics.

Yao, Y., Ye, D., Li, P., Han, X., Lin, Y., Liu, Z., & Sun, M. (2019). DocRED: A Large-scale Document-level Relation Extraction Dataset. *arXiv preprint arXiv:1906.06127.*

Zhou, G., Su, J., Zhang, J., & Zhang, M. (2005). Exploring Various Knowledge in Relation Extraction. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (ACL'05). Association for Computational Linguistics.

Zelenko, D., Aone, C., & Richardella, A. (2003). Kernel Methods for Relation Extraction. *Journal of Machine Learning Research, 3*, 1083-1106.

Zhang, S., Zheng, D., Hu, X., & Yang, M. (2015). Bidirectional Long Short-Term Memory Networks for Relation Classification. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation* (pp. 73-78). Association for Computational Linguistics.

Ni, T., Wang, Q., & Ferraro, G. (2021). Explore BiLSTM-CRF-Based Models for Open Relation Extraction. *arXiv preprint arXiv:2104.12333.*

Wu, S., & He, Y. (2019). Enriching Pre-trained Language Model with Entity Information for Relation Classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2361-2364). Association for Computing Machinery.

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning, 20(*3), 273–297.

Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Machine Learning* (pp. 139-144).

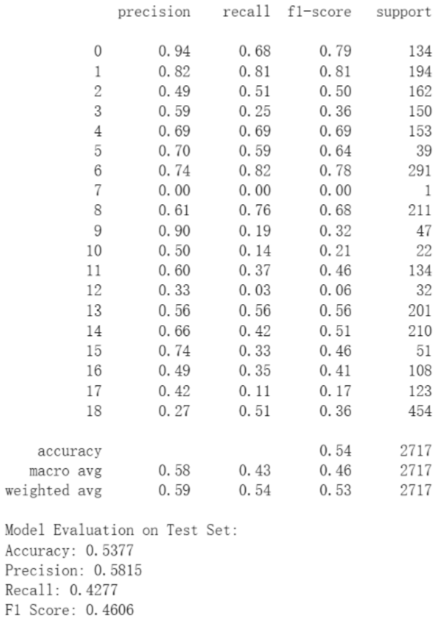
Oord, A. v. d., Li, Y., & Vinyals, O. (2018). Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748.*

Li, W., & Wu, Y. (2020). Exploiting WordNet Synset and Hypernym Representations for Answer Selection. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing.* Association for Computational Linguistics.

Sokolova, M., & Lapalme, G. (2009). A Systematic Analysis of Performance Measures for Classification Tasks. *Information Processing & Management, 45*(4), 427–437.

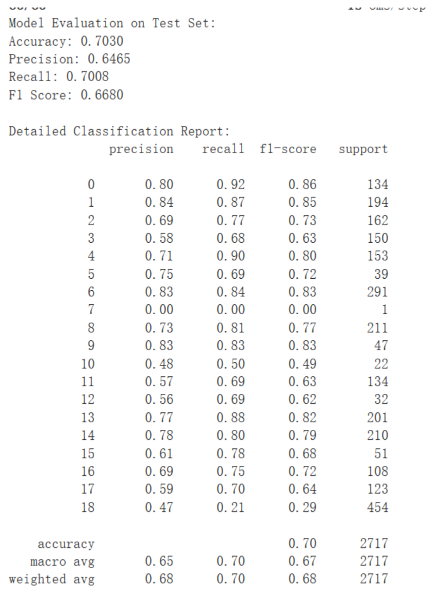
Appendix A

SVM Running results



```
              precision    recall  f1-score   support

           0       0.94      0.68      0.79       134
           1       0.82      0.81      0.81       194
           2       0.49      0.51      0.50       162
           3       0.59      0.25      0.36       150
           4       0.69      0.69      0.69       153
           5       0.70      0.59      0.64        39
           6       0.74      0.82      0.78       291
           7       0.00      0.00      0.00         1
           8       0.61      0.76      0.68       211
           9       0.90      0.19      0.32        47
          10       0.50      0.14      0.21        22
          11       0.60      0.37      0.46       134
          12       0.33      0.03      0.06        32
          13       0.56      0.56      0.56       201
          14       0.66      0.42      0.51       210
          15       0.74      0.33      0.46        51
          16       0.49      0.35      0.41       108
          17       0.42      0.11      0.17       123
          18       0.27      0.51      0.36       454

    accuracy                           0.54      2717
   macro avg       0.58      0.43      0.46      2717
weighted avg       0.59      0.54      0.53      2717

Model Evaluation on Test Set:
Accuracy: 0.5377
Precision: 0.5815
Recall: 0.4277
F1 Score: 0.4606
```

Figure A.1. Original result of SVM in SemEval 2010 Task 8 Dataset

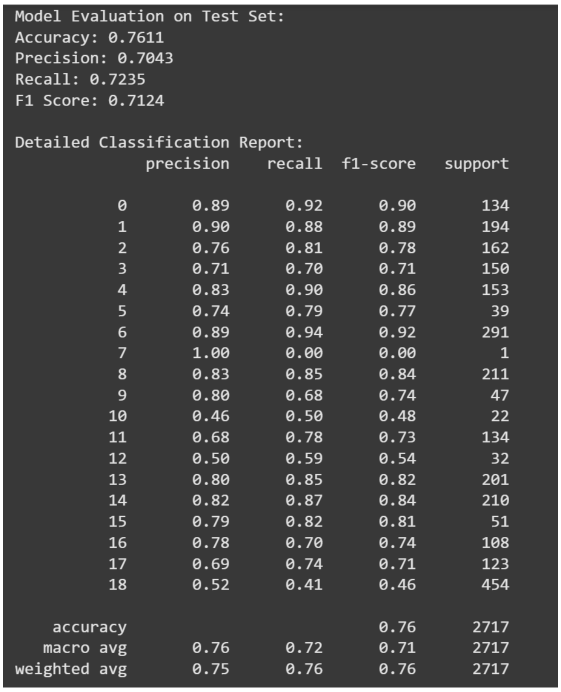


```
Model Evaluation on Test Set:
Accuracy: 0.7030
Precision: 0.6465
Recall: 0.7008
F1 Score: 0.6680

Detailed Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.92      0.86       134
           1       0.84      0.87      0.85       194
           2       0.69      0.77      0.73       162
           3       0.58      0.68      0.63       150
           4       0.71      0.90      0.80       153
           5       0.75      0.69      0.72        39
           6       0.83      0.84      0.83       291
           7       0.00      0.00      0.00         1
           8       0.73      0.81      0.77       211
           9       0.83      0.83      0.83        47
          10       0.48      0.50      0.49        22
          11       0.57      0.69      0.63       134
          12       0.56      0.69      0.62        32
          13       0.77      0.88      0.82       201
          14       0.78      0.80      0.79       210
          15       0.61      0.78      0.68        51
          16       0.69      0.75      0.72       108
          17       0.59      0.70      0.64       123
          18       0.47      0.21      0.29       454

    accuracy                           0.70      2717
   macro avg       0.65      0.70      0.67      2717
weighted avg       0.68      0.70      0.68      2717
```

Figure A.2. Optimized result of SVM in SemEval 2010 Task 8 Dataset

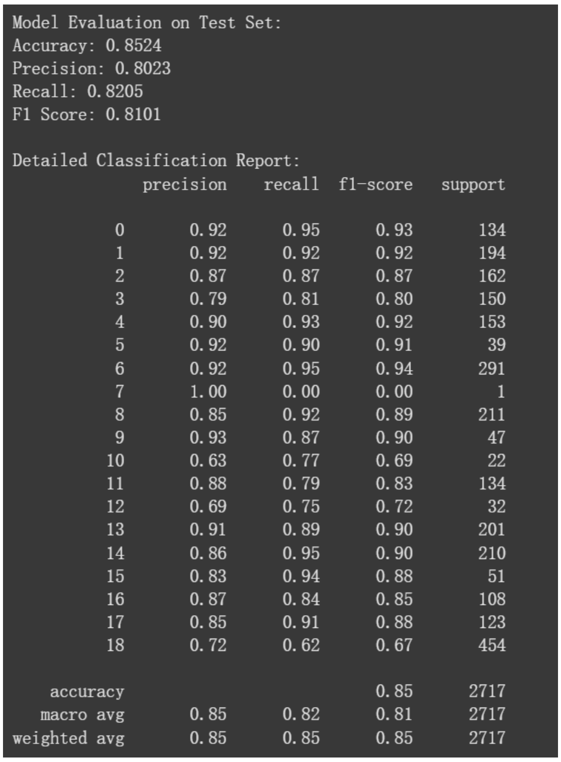
Appendix B

BERT Running results



```
Model Evaluation on Test Set:
Accuracy: 0.7611
Precision: 0.7043
Recall: 0.7235
F1 Score: 0.7124

Detailed Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.92      0.90       134
           1       0.90      0.88      0.89       194
           2       0.76      0.81      0.78       162
           3       0.71      0.70      0.71       150
           4       0.83      0.90      0.86       153
           5       0.74      0.79      0.77        39
           6       0.89      0.94      0.92       291
           7       1.00      0.00      0.00         1
           8       0.83      0.85      0.84       211
           9       0.80      0.68      0.74        47
          10       0.46      0.50      0.48        22
          11       0.68      0.78      0.73       134
          12       0.50      0.59      0.54        32
          13       0.80      0.85      0.82       201
          14       0.82      0.87      0.84       210
          15       0.79      0.82      0.81        51
          16       0.78      0.70      0.74       108
          17       0.69      0.74      0.71       123
          18       0.52      0.41      0.46       454

    accuracy                           0.76      2717
   macro avg       0.76      0.72      0.71      2717
weighted avg       0.75      0.76      0.76      2717
```

Figure B.1. Original result of BERT in SemEval 2010 Task 8 Dataset



```
Model Evaluation on Test Set:
Accuracy: 0.8524
Precision: 0.8023
Recall: 0.8205
F1 Score: 0.8101

Detailed Classification Report:
              precision    recall  f1-score   support

           0       0.92      0.95      0.93       134
           1       0.92      0.92      0.92       194
           2       0.87      0.87      0.87       162
           3       0.79      0.81      0.80       150
           4       0.90      0.93      0.92       153
           5       0.92      0.90      0.91        39
           6       0.92      0.95      0.94       291
           7       1.00      0.00      0.00         1
           8       0.85      0.92      0.89       211
           9       0.93      0.87      0.90        47
          10       0.63      0.77      0.69        22
          11       0.88      0.79      0.83       134
          12       0.69      0.75      0.72        32
          13       0.91      0.89      0.90       201
          14       0.86      0.95      0.90       210
          15       0.83      0.94      0.88        51
          16       0.87      0.84      0.85       108
          17       0.85      0.91      0.88       123
          18       0.72      0.62      0.67       454

    accuracy                           0.85      2717
   macro avg       0.85      0.82      0.81      2717
weighted avg       0.85      0.85      0.85      2717
```

Figure B.2. Optimized result of BERT in SemEval 2010 Task 8 Data

Appendix C

Project GitHub Link

https://github.com/beebatter/Text-Mining-Relation-Extraction/tree/main