Enhanced Relation Extraction by leveraging Distant Supervision in BERT and BERT-GCN Hybrid Models

Aditya Vikram Nigam[1], Tejal Ravikumar Yekkula[2], Amit Rao[3], and Sweta Das[4]

[1] *School of Engineering, 2025 University of Manchester, Manchester, M14 P10*

[*] *adityavikram.nigam@postgrad.manchester.ac.uk | tejalravikumar.yekkula@postgrad.manchester.ac.uk | amit.rao@postgrad.manchester.ac.uk | sweta.das@postgrad.manchester.ac.uk*

Relation Extraction (RE) is crucial in Natural Language Processing (NLP) for identifying semantic relationships between entities in text, supporting applications like knowledge graph construction. In this work, we propose two approaches to relation extraction. The first approach explores and experiments techniques of Distant Supervision and input representation to improve BERT (Bidirectional Encoder Representations from Transformers)-based models performance for relation extraction tasks. The second approach introduces a BERT-GCN (Graph Convolutional Networks) hybrid model, which integrates BERT's contextual embeddings with GCNs to leverage syntactic structures for improved relation classification.

1. INTRODUCTION

Relation Extraction (RE) is a fundamental task in Natural Language Processing (NLP) that enables machines to identify and classify semantic relationships between entities in unstructured text. It plays a crucial role in knowledge graph construction, information retrieval, automated reasoning, and facilitating structured knowledge extraction from vast text corpora. Traditional RE approaches have evolved from rule-based systems to deep learning models, with Transformer-based architectures like BERT achieving state-of-the-art performance.

We focus on the SemEval-2010 Task 8 [3], a widely used dataset for multi-way relation classification. The dataset comprises 10,717 annotated sentences across 19 relation types (e.g., Cause-Effect, Instrument-Agency), providing a controlled setting for evaluating RE models.

2. APPROACH A

A. Related Work

Supervised learning methods for relation classification require large amounts of labeled data, which is expensive and time-consuming to obtain. To mitigate this, distant supervision [5] was introduced as a weakly supervised learning approach that automatically generates training labels by aligning text corpora with structured knowledge bases (e.g., Freebase, Wikidata). Large-scale training datasets are possible to construct by heuristically assuming that if two entities participate in a relation in a knowledge base, then all sentences containing those entities express that relation. This has been one of the main inspirations behind our work, and we have tried to build on this approach for our chosen dataset and model.

R-BERT [8] incorporates explicit entity information, leading to improved relation classification performance. It enhances entity representations by applying separate [CLS] tokens for each entity mention, ensuring that entity embeddings capture more contextualized relational semantics rather than being overshadowed by broader sentence-level embeddings. In particular, [8] highlights distant supervision as a promising future direction, which we extend by applying DS to R-BERT and evaluating its impact on classification performance.

B. Methodology

The main objective of the project was not to beat the state-of-the-art model or provide a novel approach to relation extraction, but to perform rigorous experimentation with a combination of techniques for analysis and study in the field of relation extraction. The main contributions are summarized in this section.

B.1. Distant Supervision (DS)

In this work, we implemented DS by extracting additional training instances from the OSCAR dataset [6]. For building the knowledge base, we resolved the conflicting entity pairs (0.36%) by selecting the most frequent relation. To retrieve entities beyond exact matches, we trained a Word2Vec model to learn word embeddings, and these entity representations were used for retrieval of semantically similar entities. The initial approach of brute-force sentence retrieval ($O(N \times M)$) was computationally expensive. We implemented an optimized approach with inverted index, mapping words to sentences and retrieving only those containing either e_1 or e_2, significantly reducing comparisons. However, this required filtering to retain only those sentences where both entities appeared together.

The final training dataset consisted of approximately 16,000 sentences, doubling the original Semeval 8,000 manually labeled instances with an additional 8,000 distant supervision samples.

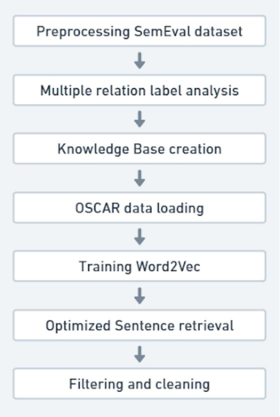
Preprocessing SemEval dataset
Multiple relation label analysis
Knowledge Base creation
OSCAR data loading
Training Word2Vec
Optimized Sentence retrieval
Filtering and cleaning

Fig. 1. Distant Supervision Process

B.2. Input representation

Standard BERT-based relation classification models use special entity markers (e.g., $E1$ and #E2#) to highlight entities, providing explicit localization. This work explored a simpler alternative, appending the entity pair at the end while preserving sentence structure. This approach would reduce special tokens, simplifying input processing in complex transformer models, which can lead to better generalization. Additionally, explicitly repeating entities reinforces their presence, helping the model focus on relationships and capture attention more effectively.

B.3. Experimental Setup

This section describes the experimentation carried out with a combination of NLP and ML techniques and different models, following the sequence of work conducted, the challenges faced, and the associated solutions.

B.3.1. Basic BERT Model

The first experiment involved fine-tuning a pre-trained BERT-base (12-layer Transformer encoder) model [1] for relation classification on the SemEval-2010 Task 8 dataset using our input representations and also with traditional $# markers.

B.3.2. Class Weighting

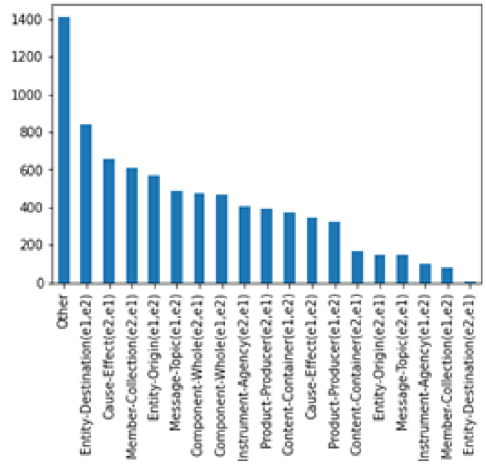

Fig. 2. Class Counts

Data analysis revealed a skewed class distribution, with *Other* comprising 20% of the dataset, while *Entity-Destination*(e_2, e_1) had only one instance. The test data showed a similar imbalance. To address this, we applied inverse frequency class weighting, ensuring underrepresented classes contributed equally to the loss function. Additionally, manual adjustments were made to prevent instability from the single-instance class.

B.3.3. R-BERT with Distant Supervision

Finally, we evaluated a reimplementation of R-BERT moodel [8] and used our DS-augmented dataset for training this model. This was a direct effort in making a step towards the future work mentioned in R-BERT research.

B.4. Final Model Architecture

The final developed model architecture integrates:

- Distant supervision (DS) for additional training data and the proposed input representation.
- A BERT tokenizer using the pre-trained 'bert-base-uncased' vocabulary.
- A BERT-base (12-layer, 768-dim, 12 heads) transformer encoder with a classification head on top containing a single linear fully connected (FC) layer.
- Class-weighted cross-entropy loss.

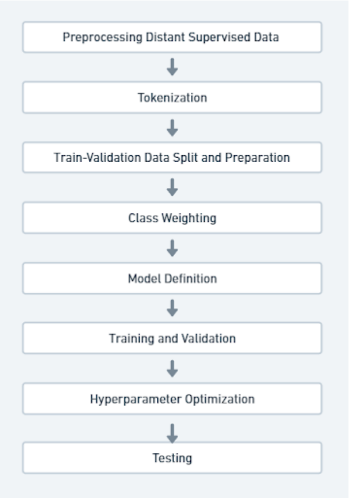
Preprocessing Distant Supervised Data
Tokenization
Train-Validation Data Split and Preparation
Class Weighting
Model Definition
Training and Validation
Hyperparameter Optimization
Testing

Fig. 3. DS-CW-Bert Approach

Hyperparameter	Value
Epochs	5
Batch Size	64
Learning Rate	7.6×10^{-5}
Weight Decay	1×10^{-8}
Epsilon	0.01
Warmup Steps	10%

Table 1. Hyperparameter settings for the developed model.

Model	Val. Acc.	Val. F1 (-O)
Baseline BERT (our input scheme)	73%	47%
Baseline BERT ($# markers)	72%	47%
BERT + DS + CW (our input scheme)	89%	71%

Table 2. Validation Accuracy and F1 Scores for Different Models

Model	Test Acc.	Test F1 (-O)
BERT + DS + CW ($# markers)	80%	55%
BERT + DS + CW (our input scheme)	80%	66%

Table 3. Test Accuracy and F1 Scores for Different Models

Model	Test Acc.	Test F1 (-O)
RBERT	83%	79%
RBERT with DS dataset	83%	80.5%

Table 4. Test Accuracy and F1 Scores for RBERT Models

C. Results and Discussion

The evaluation metrics used were accuracy and macro-averaged F1 score without the class ‘Other’, reported as F1(-O). Distant supervision significantly improves BERT’s performance, with validation accuracy increasing from 73% to 89% and F1 rising to 71% with tuning. R-BERT also benefits from distant supervision, with a F1 improvement from 79% to 80.5%. The test F1 improves by 10% when using our input representation compared to $# markers.

The proposed DS-CW-BERT model, taking around 5 minutes of training per epoch, gives a validation accuracy of 89% and a test accuracy of 80%, while R-BERT took 20+ minutes per epoch, giving a similar test accuracy. The high accuracy and quadruple reduction in training time highlight the efficiency of the DS-CW-BERT model, making it not only computationally cheaper but also more practical for real-world applications.

3. APPROACH B

A. Related Work

Relation Extraction (RE) has been extensively explored using deep learning architectures, including Convolutional Neural Networks (CNNs), Graph Convolutional Networks (GCNs), and Transformer-based models like BERT. While CNNs and Recurrent Neural Networks (RNNs) capture local and sequential dependencies, GCNs have proven effective in modeling syntactic structures by leveraging dependency trees [9].

A significant advancement in GCN-based RE is the Attention-Guided Graph Convolutional Network (AGGCN) [2], which introduces soft-pruning of dependency trees by dynamically assigning attention weights to edges. This approach enhances important syntactic relations while filtering out noise, enabling better feature propagation across multiple GCN layers. However, AGGCN relies on static GloVe embeddings, limiting its ability to adapt to context-dependent meanings.

Our GCN+BERT model addresses these limitations by integrating BERT embeddings with GCN-based relational reasoning, preserving the full dependency structure while enriching node representations with deep contextual semantics. This hybrid approach improves long-range dependency modeling, relation classification accuracy, and generalization, making it particularly effective on the SemEval-2010 Task 8 dataset.

B. Methodology

Relation extraction (RE) is a fundamental task in natural language processing (NLP) that benefits from both syntactic structure and contextual word representations. Our proposed GCN+BERT model integrates Graph Convolutional Networks (GCNs) with BERT embeddings, combining the structured learning capabilities of GCNs with the context-aware representations of BERT.Our model retains the original dependency graph while enhancing node representations with pre-trained contextual embeddings.

B.1. Contextualized Word Representations Using BERT

Traditional GCN-based RE models, such as AGGCN, rely on static word embeddings like GloVe [7], which fail to capture word sense disambiguation and contextual variability. AGGCN attempts to mitigate this limitation by introducing attention-weighted adjacency matrices, dynamically adjusting dependency relations based on their importance in the RE task. However, this modifies the original linguistic structure, potentially removing crucial syntactic dependencies. Our model addresses this limitation by employing BERT embeddings [1] as dynamic node features in the dependency graph. BERT’s transformer-based self-attention mechanism ensures that each token representation is influenced by contextual dependencies within the sentence. Unlike AGGCN, which needs to learn edge weights through self-attention mechanisms, our approach directly leverages BERT’s pre-trained representations, ensuring richer and more accurate feature extraction.

- BertTokenizer for better tokenization using subwords.
- BERT model (`bert-base-uncased`) to generate token-level hidden state representations capturing contextual semantics.
- Final hidden layer outputs as initial node feature representations in the dependency graph.

B.2. Dependency Graph Construction

Both AGGCN and our model rely on dependency parsing to construct syntactic graphs. In AGGCN, the dependency graph is transformed into an attention-weighted fully connected graph, where edge importance is learned dynamically. In contrast, our model preserves the full dependency tree as extracted using spaCy’s dependency parser [4], ensuring that the original syntactic structure is maintained. Each word is a node, and edges represent syntactic dependencies between words. Unlike AGGCN, which modifies adjacency matrices through self-attention, we retain the adjacency matrix without alterations, allowing GCNs to process the raw syntactic structure directly.

- spaCy (`en_core_web_sm`) for dependency parsing.
- Tokens as nodes; syntactic dependencies define edges.
- Adjacency matrix constructed directly from dependency relations, no attention-based re-weighting happens.
- Self-loops added to adjacency matrix to preserve word identity during GCN propagation.

B.3. Graph Convolutional Network (GCN) for Feature Propagation

Both AGGCN and our model utilize GCNs to propagate information across dependency structures. AGGCN employs multiple GCN layers with dense connections, where the attention-modulated adjacency matrix determines how features are aggregated across nodes. While this approach enhances long-range dependency modeling, it increases model complexity and introduces overfitting risks, particularly on small datasets like SemEval-2010 Task 8 [3]. Our model adopts a simpler but effective GCN-based architecture, using two GCN layers to propagate BERT-enhanced representations through dependency edges. Unlike AGGCN, which densely connects all layers, we employ batch normalization and dropout regularization to ensure stability in feature propagation and prevent over-smoothing.

- Two GCN layers with 512 and 256 hidden units for controlled feature propagation.
- ReLU activation after each GCN layer for non-linearity.
- Batch normalization after each layer to stabilize gradient flow.
- Dropout (0.3) to prevent overfitting and enhance generalization.

B.4. Relation Classification Using Pooled Representations

AGGCN classifies relations by concatenating entity representations with sentence embeddings derived from attention-weighted GCN outputs. Our model simplifies this process by applying global average pooling over GCN outputs, ensuring that all tokens contribute to relation classification without introducing additional attention layers. This approach ensures efficient feature aggregation while maintaining structural integrity.

- Global average pooling on GCN outputs to aggregate sentence-level representations.
- Fully connected layer maps pooled representations to relation labels.

C. Results and Discussion

Our GCN+BERT model achieves high classification accuracy across all three datasets—train, validation, and test. The training set achieves an accuracy of 96.78% and a macro-F1 score of 91.29%, indicating strong learning capabilities. The validation set maintains a similarly high accuracy of 97.19%, confirming effective generalization.

However, performance on the test set drops to 83.29% accuracy and a macro-F1 score of 79.21%, highlighting challenges in handling unseen data. The confusion matrices reveal that misclassifications are more pronounced in relation types with fewer training samples, particularly in cases where semantic overlap or syntactic ambiguity exists.

A deeper analysis of the test set confusion matrix shows that relations such as Relation-18 (Cause-Effect) and Relation-10 (Content-Container) exhibit higher misclassification rates. Relation-18, which had the largest number of test samples (454), was correctly classified 266 times but frequently misclassified across multiple categories, leading to a F1-score of only 62.37%. Similarly, Relation-10 (Content-Container) struggles with low recall (63.64%), suggesting that the model confuses container-based relationships with other spatial relations. These errors often arise when dependency structures do not clearly separate the entities, limiting the effectiveness of graph propagation. The model also fails to classify Relation-7, potentially due to its low frequency in the dataset, leading to zero recall for this category.

The error analysis report provides further insights, revealing that 454 errors occurred in the test set, translating to an overall test error rate of 16.71%. Common error patterns include misclassifications between semantically similar relation types (e.g., Component-Whole vs. Member-Collection) and inconsistent entity spans that affect dependency graph quality. Additionally, errors in longer sentences with multiple relations suggest that BERT embeddings may not fully disambiguate overlapping dependencies. While GCN+BERT effectively leverages syntactic structure, its performance is sensitive to dependency parsing quality, indicating potential areas for improvement in graph representation strategies. Future enhancements could include graph attention mechanisms to prioritize more relevant dependencies or contrastive learning techniques to better distinguish closely related relation classes.

Dataset	Accuracy (%)	Macro-F1 Score (%)	Error (%)
Training	96.78	91.29	3.22
Validation	97.19	91.95	2.81
Test	83.29	79.21	16.71

Table 5. Performance metrics of the GCN+BERT model across different datasets.

Hyperparameter	Value
Learning Rate (BERT)	1e-5
Learning Rate (GCN)	2e-5
Optimizer	AdamW
Dropout Rate	0.3
Batch Size	16
Number of GCN Layers	2
Hidden Layer Dimensions	512, 256
Activation Function	ReLU
Epochs	5
Learning Rate Scheduler	ReduceLROnPlateau
Early Stopping Patience	3

Table 6. Hyperparameter configuration of the GCN+BERT model.

4. CONCLUSION

The study shows that While distant supervision introduces label noise due to ambiguous entity pairs appearing in multiple contexts, this diversity helps the model generalize by learning robust decision boundaries, potentially reducing overfitting. Input representation impacts both accuracy and efficiency, and the entity appending approach offers advantages over traditional entity markers. Beyond data augmentation and input design, incorporating structural information through Graph Convolutional Networks (GCNs) strengthens relation extraction. By integrating BERT's contextual embeddings with syntactic graphs, our hybrid model improves relational reasoning, though its effectiveness depends on dependency parsing quality. The results highlight that these techniques can enhance BERT models in relation classification and hold strong potential for future research in Natural Language Processing.

REFERENCES

[1] J. Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *arXiv* (2019). arXiv: 1810.04805.

[2] Z. Guo, Y. Zhang, and W. Lu. “Attention Guided Graph Convolutional Networks for Relation Extraction”. In: *Proceedings of ACL*. 2019.

[3] I. Hendrickx et al. “SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations Between Pairs of Nominals”. In: *Proceedings of ACL*. 2010.

[4] M. Honnibal and I. Montani. *spaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks, and Incremental Parsing*. GitHub. 2017.

[5] M. Mintz et al. “Distant Supervision for Relation Extraction without Labeled Data”. In: *Proceedings of ACL*. 2009.

[6] Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. “A Monolingual Approach to Contextualized Word Embeddings for Mid-Resource Languages”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 1703–1714.

[7] J. Pennington, R. Socher, and C. Manning. “GloVe: Global Vectors for Word Representation”. In: *Proceedings of EMNLP*. 2014.

[8] Shanchan Wu and Yifan He. “Enriching Pre-trained Language Model with Entity Information for Relation Classification”. In: *arXiv preprint arXiv:1905.08284* (2019). Accessed: 2025-03-06.

[9] Y. Zhang, P. Qi, and C. D. Manning. “Graph Convolution over Pruned Dependency Trees Improves Relation Extraction”. In: *arXiv* (2018). arXiv: 1806.08992.

GENERATIVE AI DISCLOSURE

OpenAI’s ChatGPT and ***Deepseek*** were utilized in limited manner, exclusively for refining linguistic structure and improving readability. The core methodology, ideas, logic, and reasoning presented in this work were developed entirely independently by the authors.