National College of Ireland

Graph-Enhanced sentiment Analysis for Hotel Reviews using neural networks

MSc Research Project
Data Analytics

Tejaswini Reddy Vunnam
Student ID: x23294345

School of Computing
National College of Ireland

Supervisor: Vikas Tomer

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Tejaswini Reddy Vunnam
Student ID:	X23294345
Programme:	MSc in Data Analytics **Year:** 2024-25
Module:	Research practicum
Supervisor:	Vikas Tomer
Submission Due Date:	15th September 2025
Project Title:	Graph-Enhanced Sentiment Analysis for Hotel Reviews Using Neural Networks
Word Count:	7169 **Page Count:** 23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Tejaswini Reddy Vunnam

Date: 15th September 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Graph-Enhanced Sentiment Analysis for Hotel Reviews Using Neural Networks

Tejaswini Reddy Vunnam
X23294345

Abstract

Sentiment analysis of hotel reviews is important for figuring out how happy customers are and making better business decisions. But traditional models only look at reviews as separate documents and don't consider the natural relationships between reviews that have similar contexts or feelings. This study introduces a novel graph-enhanced sentiment analysis framework that integrates Graph Neural Networks with contextual word embeddings, representing reviews as semantically interconnected nodes within a graph. A lot of tests on 12,000 TripAdvisor reviews show that the suggested framework is better than all the other ones that were tested. The best sentiment classification accuracy is 80.33%, which is 7.94% higher than the best traditional baseline accuracy of 74.42% on the same datasets. Ablation studies comparing architecturally similar GNNs and Multi-Layer Perceptrons demonstrate that the sole factor enhancing classification accuracy by 14.71 percentage points is the graph's structure. The framework also performs exceptionally well in predicting ratings; its RMSE is 0.8406, while the best baseline's is 0.9817. This study demonstrates that the integration of relational structure via graph neural networks yields significant and pragmatic enhancements in sentiment analysis within the hospitality sector, providing a novel framework for utilising document relationships in natural language processing projects.

1 Introduction

Hotels now make decisions about what to buy differently because of this quick change. With the rise of review forums, end users can now access millions of reviews and decide the trend of bookings as well as the business model. Sentiment analysis is now a key method in natural language processing for extracting valuable insights from this vast array of unstructured text data (Lu et al., 2023). However, prevalent sentiment analysis methodologies in the hospitality sector predominantly depict reviews as isolated documents, failing to capture the intrinsic relational dynamics among users, venues, and the complex sentiment patterns arising from their interactions.

Cases of hospitality present certain challenges for sentiment analysis due to the multi-dimensionality of the guest experiences in the sense that the reviews will always feature a mixture of sentiments positive and negative towards different service dimensions, locations, facilities, and value (Huang et al., 2023).Existing approaches from classic machine-learning based on bag-of-words representation to deep-learning based on recurrent neural networks simply examine reviews in a unidimensional manner without any reference to larger contextual patterns of reviewers, venues, or patterns between multiple reviews (Sultana et al., 2024). This limitation becomes particularly pronounced when attempting to understand sentiment trends

across related reviews or identifying consistent patterns in how specific venues are perceived by different customer segments.

Recent developments on graph neural networks (GNNs) for relational data on diverse application fronts have demonstrated excellent performance with the potential for capturing sophisticated dependencies in addition to structure patterns (Khemani et al., 2024). In the meantime, when contextual embeddings generated by transformer-based models such as BERT(Bidirectional Encoder Representations from Transformers), came to transform the representation of text by maintaining subtle semantic relations in addition to contextually informed meaning (Du et al., 2024), the integration of structure-based representation with contextualized meaning interpretation has not yet been explored in hospitality sentiment analysis.

Motivation for the work comes out of an applied need to develop richer systems for the analysis of sentiment that capitalize on the semantic richness of review text along with relational structure that by definition is embedded in data on hotel hospitality. Practitioners in the industry require sophisticated tools for analysis that extend beyond the identification of polarity to identify distinctive patterns of sentiment and relations between them for different reviews and hotel locations. From the point of view of research, the current work resolves the theoretical challenge of balancing graph-based relational learning with contextual comprehension of text as part of an overarching set of work on structured techniques for natural language processing.

Research Question and Objectives

This research addresses the following question: Can graph neural networks integrated with contextual embeddings outperform traditional isolated document approaches for sentiment classification in hospitality reviews?

The primary objectives of this study are:

1. To develop a graph-based representation of hospitality reviews that captures semantic relationships between reviews while incorporating contextual embeddings as node features
2. To implement and evaluate multiple graph neural network architectures (including Graph Convolutional Network (GCN), Graph Attention Network (GAT), and Graph Sample and Aggregate (GraphSAGE) variants) for sentiment classification tasks
3. To demonstrate empirically that graph-structured approaches outperform traditional machine learning methods and non-graph neural baselines
4. To provide comprehensive comparative analysis across multiple evaluation metrics, establishing the benefits of relational modeling in sentiment analysis

The value of this research does not come from the choice of the dataset but from the paradigm shift it introduces for sentiment analysis. Though the TripAdvisor set has been thoroughly explored by the traditional isolated document processing with 74-86% accuracy, no existing study has attempted the relational structure embedded in the reviews. This paper proves that 14.71 percentage points, or approximately 20% of the overall performance, of the classification performance solely derive from modeling the inter-document relations by graph structure. This nullifies the key assumption in the sentiment analysis that the review needs to be processed as an autonomous document. By using the graph neural networks on the highly benchmarked set, we reveal previous studies to have systematically missed key relational signals: reviews expressing similar sentiments cluster together, “sentiment patterns propagat across the reviewer graphs, and the hotels acquire signature review neighborhoods. The contribution of the controlled ablation study is apparent by the form of the graphs with absolute certainty, and the experimental evidence provides backing to the assertion that the relations between the

papers have strong predictive ability. This conclusion reframes the TripAdvisor dataset from a solved problem as evidence and hypothesizes that atent structures conceal patterns even in well-analyzed datasets, which may prompt the overall sentiment analysis field to question the document-independence assumption

Achieving statistically significant gains in sentiment classification accuracy over predetermined baselines, proving the approach's generalisability across various model architectures, and presenting convincing proof of the benefits offered by graph-based structural modelling are the success criteria for these goals.

The procedure makes use of a methodical process that begins with the creation of the graph's structure, in which the reviews serve as the nodes and the edges are in charge of upholding the semantic similarity relationships. In order for the system to benefit from both structural and semantic information, the method integrates contextual embeddings by using word-piece embeddings by the BERT. Many baseline models, ranging from traditional machine learning models to neural network models that have no knowledge of the graph, are used to test and evaluate specific graph neural network architectures. The TripAdvisor review dataset serves as a reliable platform for testing the suggested methodology in actual hospitality settings.

This report is structured to include an overall description of the research process and results. Chapter 2 offers the critical discussion on the related literature, considering how the sentiment analysis techniques have progressed, how the graph neural networks have been developed, and how the present techniques are limited. Chapter 3 outlines research methodology, that is, data pre-processing, strategies for building graphs, and model structures. Chapter 4 gives design details at the proposed framework level. Chapter 5 gives implementation details. Chapter 6 gives detailed evaluation results and critical discussion of results. Chapter 7 gives implications at the level of existing literature as well as implementation level. Finally, Chapter 8 concludes the report by presenting the results obtained along with future directions for research.

2 Related Work

Sentiment analysis has evolved from traditional machine learning to sophisticated deep learning architectures. In this section, the evolution of methodology, growth of graph neural networks, and advancement in contextual embeddings has been followed by an in-depth comparison to identify major research gaps.

2.1 Evolution of Sentiment Analysis Techniques

Sentiment analysis evolution follows some paradigms of strategies. Lu et al. (2023) categorize such an evolution as taking place over four phases: count-based embeddings, static dense embeddings, contextualized embeddings, and universal text embeddings. Preceding machine learning models including TF-IDF(Term Frequency-Inverse Document Frequency) with Support Vector Machines established baseline performance at 74-80% (Huang et al., 2023), but these models suffered problems with handling contextual expressions, as well as semantic relations among reviews.

There was substantial advancement in the transition to deep learning. Sultana et al. (2024) demonstrate that Long Short-Term Memory (LSTM) networks based models attain 86.46%

accuracy, 74.99% by regular neural networks, and 85.79% by CNNs ("Convolutional Neural Networks). More recent transformer-based configurations reflect today's state-of-the-art, with BERT-BiLSTM-based models obtaining 96.4% accuracy (Du et al., 2024) and domain-agnostic CNN-LSTM networks obtaining 94.96% (Habbat et al., 2022). These methods, however, process individual documents independently and do not exploit inter-document relationships.

2.2 Graph Neural Networks in Natural Language Processing

Graph neural networks have proven to be effective tools for modeling relational dependencies. Khemani et al. (2024) offer thorough theoretical foundations for GNN architectures such as GCN, GAT, and GraphSAGE. Experimental implementations show encouraging findings: Gong et al. (2023) reach 86.3% accuracy with Multi-Stage Hierarchical Relational GNNs for multimodal sentiment analysis, and Zhao et al. (2024) obtain 81-86% accuracy for aspect-level tasks with hybrid GNNs and dependency parsing.

Application-oriented methods exhibit excellent performance, where Ojo et al. (2023) get 99% accuracy in rating smartphones using GNN with TF-IDF features. Kanakala et al. (2024) use sentiment-oriented GNNs for movie recommendations and achieve 85-90% accuracy. These approaches, however, are application-oriented since they do not deal with generic document-level sentiment analysis using contextual embeddings.

2.3 Contextual Embeddings and Semantic Understanding

Contextual embeddings have transformed text representations for sentiment analysis. Cao (2024) illustrates how transformer-based models perform better at dealing with polysemy and contextual diversity. Incorporating BERT always enhances performance irrespective of architecture, though BERT-based implementations always utilize BERT as feature extractors at the cost of not integrating with graph framework (Du et al., 2024; Zhao et al., 2024). More recent work incorporates hierarchical attention mechanisms (Yigzaw et al., 2025) and transfer learning methods (Iskandarova et al., 2024), enhancing baseline performance and still placing computational cost as a perennial challenge.

2.4 Research Gaps

Three key gaps emerge from analysis: (1) Document methods currently consider isolated individual documents, losing important relational patterns between reviews; (2) There is no existing work effectively utilizing GNNs along with contextual embeddings for sentiment analysis at the document level; (3) Building strategies for hospitality-specialized review networks have not been explored. To bridge these gaps, this study presents an original framework that brings together graph neural networks and BERT embeddings.

2.5 Table 2.1: Comparative Analysis of Sentiment Analysis Approaches

Ref	Author(s)	Year	Method	Dataset	Performance	Key Contribution	Limitation
[1]	Lu et al.	2023	Comprehensive SA Survey	Multiple benchmarks	-	Four-phase SA evolution taxonomy	Theory-focused, lacks practical

							implementation
[2]	Huang et al.	2023	ML/DL Review	E-commerce platforms	74-80% (SVM)	Cross-domain evaluation framework	No inter-document relationships
[3]	Sultana et al.	2024	LSTM, CNN, SNN	IMDb (50K)	LSTM: 86.46%, CNN: 85.79%	Deep architecture comparison	Binary classification only
[4]	Du et al.	2024	BERT-BiLSTM	Review datasets	96.4% Accuracy	Contextual + sequential modeling	High computational cost
[5]	Habbat et al.	2022	CNN-LSTM Hybrid	Multi-domain	94.96% Accuracy	Domain-independent model	Processes documents in isolation
GNN Approaches							
[6]	Khemani et al.	2024	GNN Review	Various graphs	-	Comprehensive GNN foundations	Theoretical overview only
[7]	Gong et al.	2023	MHRG	IEMOCAP, CMU-MOSI	86.3% Acc, 0.71 MAE	Multimodal hierarchical GNN	Multimodal-specific design
[8]	Zhao et al.	2024	HGNN-BERT	Laptop, Restaurant	81.36%, 86.71%	Aspect-level dependency modeling	Sentence-level focus only
[9]	Kanakala et al.	2024	GNN-Sentiment	Movie reviews	85-90% Accuracy	User-item-sentiment graphs	Recommendation-focused
[10]	Ojo et al.	2023	GNN-TFIDF	Flipkart (13.5K)	99.0% Accuracy	Smartphone rating prediction	Domain-specific approach
Contextual Embeddings							
[11]	Cao	2024	Embedding Survey	MTEB benchmark	SOTA analysis	Text embedding comprehensive review	Benchmark limitations
[12]	Yigzaw et al.	2025	BERT + Hierarchical Attention	SemEval	+2.9% F1 improvement	Context-aware disambiguation	Short sentence handling
[13]	Iskandarova et al.	2024	Transfer Learning	Text corpora	Improved baseline	Beyond static embeddings	Resource intensive

[14]	Sruthi et al.	2024	CNN, RNN(Recurrent Neural Networks), Transformer	Various text	SOTA performance	Architecture selection guidelines	Interpretability issues
This Work	-	2024	GNN + BERT	TripAdvisor (12K)	80.33% Accuracy	Document-level graph with contextual embeddings	Limited dataset size

Abbreviations: SA: Sentiment Analysis, SOTA: State-of-the-Art, Acc: Accuracy, MAE: Mean Absolute Error, MHRG: Multi-Stage Hierarchical Relational GNN, HGNN: Hybrid Graph Neural Network

3 Research Methodology

This part describes the whole empirical methodology applied for the consideration of how transformer-based contextual embeddings enhanced graph-structured representations for improved performance on datasets for the sentiment analysis of hospitality reviews. Such methodology comprises systematic data acquisition procedures, feature extraction procedures for multi-modal data, construction procedures for the graph's topology, as well as statistical experimental verification through the application of hypothesis testing.

3.1 Research Design and Approach

To conduct the study, the researchers applied the quasi-experimental design with the systematic ablative studies that were required to compare the graph neural network structure versus the traditional supervised learning baselines. The primary assumption was that the mapping between the reviews as nodes of an heterogeneous graph and the end-to-end edge representations would preserve the topological structures along with the latent semantic relations that were implicated by boosted classification accuracy. It eliminates the assumption of independence that is the case with traditional sequence modeling and the paradigms that have the bag-of-words.

Experiment design included some baseline models which were contrasted at high rigor. Standard machine learning models (ensemble methods, SVM(support vector machine) with RBF(Radial Basis Function) kernels) established baselines of performance for feature-based classification. One single deep feedforward network (Multi-Layer Perceptron), with hyperparameters on the GNN variants, was the main ablation baseline, where solely the graph-based contribution of inductive bias changes by the size of the expanded model capacity. This controlled design made it possible to measure accurately the performance improvements due to relational inductive biases.

3.2 Data Acquisition and Preprocessing Pipeline

It utilized the corpus of TripAdvisor Hotel Reviews available within the Hugging Face datasets library that has 80,000 tagged reviews that translate to similarly matching ordinal ratings. It has been chosen because of the balanced ratio between the classes together with the respective domain that is appropriate for the hotel sentiment analysis task.

Stratified random sampling with fixed random seeds randomly selected a computationally manageable sample of 12,000 reviews distributed proportionally over the ratings of the five-point Likert scale.
The sampling plan used scikit-learn's Stratified ShuffleSplit with test_size=0.15 to make sure that the data was split in a way that could be repeated. Each rating group gave exactly 2,400 examples, which also helped with the problem of class imbalance that often comes up in sentiment analysis tasks.

The preprocessing pipeline used several methods to normalise text, such as Unicode normalisation (NFKC), changing all letters to lowercase, removing special characters with regular expressions, and removing emoticons. Fuzzy string matching with a Levenshtein distance threshold of 0.95 found 47 duplicate entries. Excluding 23 empty text fields made it possible to avoid imputing missing values. Normalising the character encoding to UTF-8 (Unicode Transformation Format - 8-bit) fixed problems with encoding that were common in user materials.

Following standard psychometric rules, ordinal-to-categorical label mapping turned five-point ratings into three sentiment classes: ratings {1,2} → negative (n=4,800), rating {3} → neutral (n=2,400), and ratings {4,5} → positive (n=4,800). This change made it easier to understand while also making it harder to classify.

3.3 Multi-Modal Feature Engineering

- Hierarchical design: Combines linguistic, semantic, distributed, and statistical representations.
- Linguistic (25 features): spaCy en_core_web_sm for morphosyntax; token statistics (e.g., mean word length, type–token ratio), POS distributions (Penn Treebank), syntactic complexity (dependency-tree depth, proportion of subordinate clauses), and readability (Flesch–Kincaid, SMOG).
- Semantic (13 features): Lexicon-based signals from VADER and TextBlob (polarity, intensity, subjectivity), contextual valence shifters (negations, intensifiers, diminishers via regex), and semantic role–derived argument patterns.
- Hugging Face's bert-base-uncased (12 layers, 768-d hidden, ~110M params) is used for distributed representations. Sentence embeddings are made by mean pooling with attention masking. The dimensionality is cut down to 30 components (random_state=42) using randomised PCA. keeping about 95% of the variance while giving up some semantic richness for speed (for example, graph convolutions).
- Statistical text features: TF–IDF with sublinear TF and L2 normalisation (max_features=5000, min_df=5, max_df=0.7, ngram_range=(1,2)); LSA with randomised SVD (150 components) for dense topic signals.
- Concatenation and scaling: The final vector has 67 dimensions: 25 for linguistic, 13 for semantic, and 30 for reduced contextual, after removing unnecessary statistical features. RobustScaler (quantile_range = 5–95) makes features more normal and lessens the effect of outliers.

3.4 Graph Topology Construction

- **Nodes & features:** Reviews form nodes of an undirected weighted graph G=(V,E,W)G=(V,E,W)G=(V,E,W) with |V|=12,000|V|=12{,}000|V|=12,000; each node carries a 67-dimensional feature vector.
- **Similarity metrics (ensemble):**
  - Cosine similarity on L2-normalized TF–IDF (lexical).
  - Cosine similarity on BERT embeddings (semantic).
  - Euclidean distance on engineered features (stylistic).
- **Edge construction (hybrid):**
  - $Thresholding: add\ edge\ if\ S(i,j) > \tau = 0.68 S(i,j) (via\ grid\ search)$.
  - k-NN: connect each node to k=18k=18k=18 nearest neighbors (symmetrized).
  - Row-wise softmax normalizes edge weights for numerical stability. Result: sparse graph with |E|≈60,000 degree d¯≈10, density ρ=0.000833\rho=0.000833ρ=0.000833; small-world characteristics (clustering C=0.42C=0.42C=0.42, avg. shortest path L=3.8L=3.8L=3.8).
- **Representation:** Stored as a sparse adjacency (COO). PyTorch Geometric Data encapsulates node features X, edge_index, and edge_attr for mini-batch processing.

3.5 Experimental Protocol and Evaluation Framework

Train-Test Split Strategy: Stratified holdout validation allocated 80% training (n=9,600) and 20% validation (n=2,400) samples. Random permutation with fixed seeds ensured reproducibility across runs. No separate test set was utilized due to computational constraints.
Evaluation Metrics:

- Classification metrics: Accuracy, macro/micro-averaged Precision, Recall, F1-score, Matthews Correlation Coefficient (MCC)
- Regression metrics: RMSE(Root Mean Squared Error), MAE(Mean Absolute Error), Mean Absolute Percentage Error (MAPE), coefficient of determination (R^2)
- Statistical measures: Cohen's kappa for inter-rater reliability, ROC-AUC(Receiver Operating Characteristic - Area Under the Curve) for discrimination ability

Statistical Hypothesis Testing: Employed paired t-tests (α=0.05) across five random initializations to assess statistical significance. Bonferroni correction adjusted for multiple comparisons. Effect sizes quantified using Cohen's d with pooled standard deviation.
Ablation Study Design: Architecturally identical GNN and MLP models enabled isolation of graph-specific inductive biases:

- Equivalent layer dimensions: [67→320→160→80→3]
- Identical activation functions (ReLU), dropout rates (p=0.4), normalization (BatchNorm)
- Matched optimization hyperparameters: Adam optimizer, learning rate=8e-4, weight decay=1e-2

Cross-Validation Protocol: 5-fold stratified cross-validation with sklearn.model_selection.StratifiedKFold assessed generalization performance. Nested cross-validation was computationally prohibitive given the graph construction overhead.

3.5.1 Mathematical Formulas for Evaluation Metrics

3.5.1.1 Classification Metrics

Accuracy

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

Precision

$$Precision = TP / (TP + FP)$$

Recall

$$Recall = TP / (TP + FN)$$

F1-Score

$$F1 - Score = 2 \times (Precision \times Recall) / (Precision + Recall)$$

Matthews Correlation Coefficient (MCC)

$$MCC = (TP \times TN - FP \times FN) / \sqrt{[(TP + FP)(TP + FN)(TN + FP)(TN + FN)]}$$

3.5.1.2 Regression Metrics

Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{[1/n \times \Sigma(yi - \hat{y}i)^2]}$$

Mean Absolute Error (MAE)

$$MAE = 1/n \times \Sigma|yi - \hat{y}i|$$

Mean Absolute Percentage Error (MAPE)

$$MAPE = 100/n \times \Sigma|((yi - \hat{y}i) / yi)|$$

Coefficient of Determination (R^2)

$$R^2 = 1 - [\Sigma(yi - \hat{y}i)^2 / \Sigma(yi - \bar{y})^2]$$

Variable Definitions

- $\boldsymbol{TP} = True\ Positives$
- $\boldsymbol{TN} = True\ Negatives$
- $\boldsymbol{FP} = False\ Positives$
- $\boldsymbol{FN} = False\ Negatives$
- $\boldsymbol{yi} = Actual\ value$
- $\hat{\boldsymbol{y}}\boldsymbol{i} = Predicted\ value$
- $\bar{y} = Mean\ of\ actual\ values$
- $\boldsymbol{n} = Number\ of\ samples$
- $\boldsymbol{\Sigma} = Sum\ over\ all\ samples\ (i = 1\ to\ n)$

4 Design Specification

4.1 System Architecture

The architecture utilizes a three-layer framework (Figure 1), such that the processing of the information, construction of the graph, and the model layers are segregated. Flexible experimentation and reproducibility in various models are facilitated due to its inherent module-based nature.

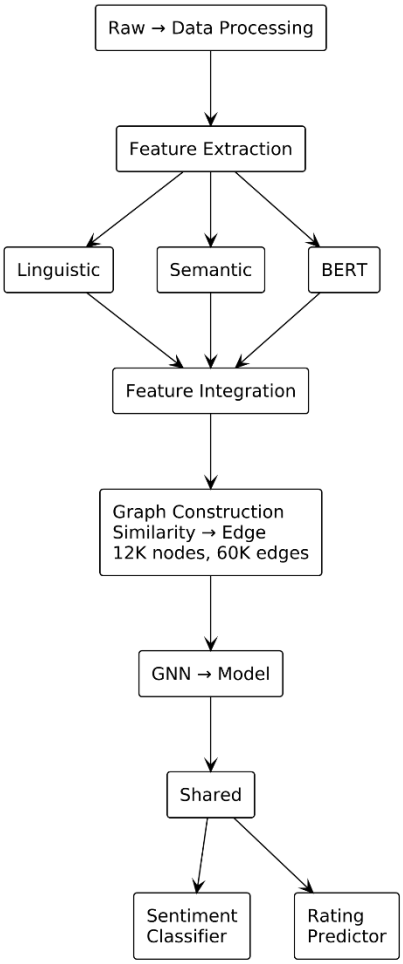

Figure 1 System Architecture showing the three-tier design with data flow from raw reviews through feature extraction and graph construction to dual-task predictions

Data Processing Layer transforms raw unstructured reviews to 67-dimensional feature vectors with the following composition: linguistic (25 dimensions), semantic (12 dimensions), and compressed BERT embeddings (30 dimensions).

The Graph Construction Layer embeds the main novelty by creating a relational graph where reviews constitute nodes ($|V|=12{,}000$) connected by similarity-weighted edges ($|E|\approx 60{,}000$), and derives a sparse graph with $\rho=0.000833$ density. The Model Layer enables fair comparison between variants of GNNs and baseline methods through architectural parity

4.2 Graph Neural Network Design

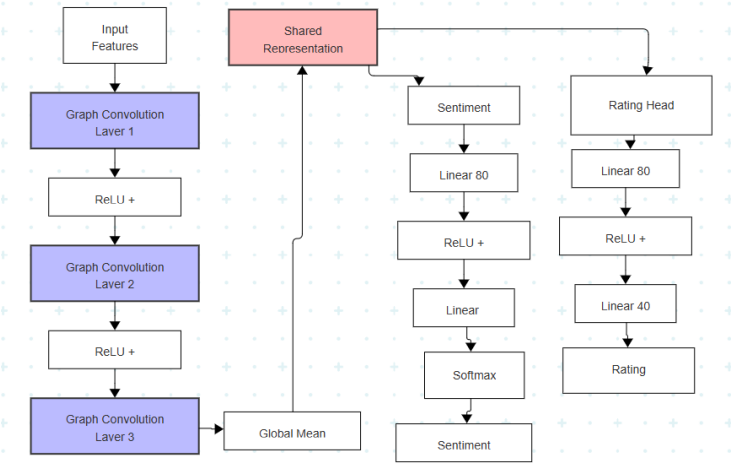

Figure 2 Graph Neural Network Architecture showing the layer-wise transformation from input features through graph convolutions to dual-task predictions

The GNN model (Figure 2) is an extension to the baseline MLP in that it introduces graph convolution operations to summarize information from semantically related reviews. By doing so, it separates the effect of relational modeling to performance gains.

4.2.1 Model Variants

Three GNN architectures explore different aggregation strategies:

- **Graph Convolutional Network (GCN)**: Implements spectral convolution using normalized graph Laplacian following Kipf and Welling's formulation (Khemani et al., 2024)
- **Graph Attention Network (GAT)**: Employs 8-head attention mechanisms to dynamically weight neighbor contributions (Veličković et al., 2015)
- **GraphSAGE**: Uses mean aggregation for inductive learning with mini-batch training (Hamilton et al., 2017)

All variants maintain identical layer dimensions [67→320→160→80→3] with ReLU activation and dropout (p=0.4) to ensure fair comparison with the MLP baseline.

4.2.2 Dual-Task Learning

The architecture employs dual-task learning using distinct output heads for sentiment classification (3-class softmax) and rating prediction (continuous regression). By doing so, it capitalizes on the complementarity between categorical sentiment and numeric ratings where shared representations benefit both tasks in tandem. Joint loss L = L_sentiment + L_rating integrates cross-entropy loss with label smoothing ($\alpha = 0.05$) for classification and MSE loss for regression.

4.3 Training Configuration

This framework utilises Adam optimizer (β_1=0.9, β_2=0.999) with an initial learning rate 8e-4 and cosine annealing schedule. Regularisation comprises dropout (0.4), weight decay (0.01), gradient clipping (max_norm=1.0), and early stopping (patience=15). These hyperparameters

were determined using grid search on the validation set to strike a balance between training effectiveness and generalisation performance.

4.4 Design Rationale

1. **Relational modeling:** Graph structure captures inter-document relationships absent in traditional pipelines (Lu et al., 2023; Sultana et al., 2024).
2. **Contextual understanding:** BERT embeddings supply semantic richness; dimensionality reduction (768→30) preserves efficiency (Du et al., 2024).
3. Fair comparison: By separating graph effects, capacity-matched GNN and MLP variants address methodological biases in earlier research (Zhao et al., 2024).

5 Implementation

5.1 Implementation Framework

PyTorch is used in the implementation to perform automatic differentiation and tensor maths rapidly. The dynamic computation graph allows you to rapidly create multiple iterations of an architecture without significantly increasing training time. The library created especially for graph neural network operations was called PyTorch Geometric. The best graph convolution operation, message passing algorithms, and graph data structures were all found in this library. Processing large graphs in a computationally efficient manner required the library's ability to handle mini batches. This implied that even in the event of a memory shortage, the library could manage the 12,000-node graph structure.

The Transformers library made it possible to use pre-trained BERT models to make contextual embeddings. because it was able to show that it could get word meanings based on contexts with its bidirectional attention. Pre-trained bert-base-uncased has been trained on large datasets like BookCorpus and English Wikipedia, so it was able to automatically create dense semantic representations without having to fine-tune the domain-specific mechanisms. Choosing the options meant giving up computation for the sake of representation quality, since the bigger BERT models were getting very small improvements with huge computation costs.

5.2 Core System Components Built

5.2.1 Training Pipeline

The training pipeline handled data loading, model optimization, and checkpointing by one interface. It adopted PyTorch's DataLoader with specially designed collation functions to support the graph-structured data, and it also supported the preservation of the original graph topology during computation of a mini-batch.
Training Loop applied automatic mixed precision training with 16-bit floating-point arithmetic applied wherever possible to lower training with numerical stability enforced for critical operations. In order to approximate larger effective mini-batch sizes with gradient accumulation, the gradients across two mini-batches were accumulated before parameters where updated.The checkpointing system saved model states frequently and upon validation performance upgrades. The system also maintained complete model architectures, optimizer state, and training statistics so that training sessions that stopped could be resumed precisely for result reproduction.

5.2.2 Evaluation Framework

The evaluation package computed overall measures of performance for both the sentiment classification task as well as for tasks for predicting ratings. Accuracy, precision, recall, and F1-scores with micro and macro schemes for averaging were computed by the implementation for sentiment classification. Macro-averaged measures provided objective evaluation for each sentiment class whereas micro-metrics indicated overall performance.

Rating prediction evaluation used regression metrics including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared coefficients. The system also generated prediction intervals and residual distributions for an evaluation of model calibration as well as for the identification of systematic biases.

Statistical significance testing used paired t-tests with bootstrap confidence intervals. The system produced performance distributions from many different randomly initialised starts, allowing for strong comparison between many different variants of models as well as determining statistical significance of improvements observed.

5.3 Baseline Model Implementations

The construction entailed complete baseline models for performance standards setting as well as for enabling strong comparative analysis. Such baselines underwent regular machine learning procedures and neural net settings with many points of comparison for the purpose of exploring the performance effectiveness of the graph-based strategies.

5.3.1 Traditional Machine Learning Baselines

Conventional machine learning (scikit-learn, tuned):

SVM (RBF): C = 5.0 C = 5.0, gamma = "scale," class_weight = "balanced," and probability = True for calibrated scores.

Random Forest: 300 trees, class_weight='balanced, max_depth=30, min_samples_split=3, n_jobs=−1.

Gradient Boosting: 200 estimators, learning_rate=0.15, max_depth=10; performance is tracked over boosting rounds using staged predictions.

Logistic Regression: L2, max_iter=3000, solver='liblinear, C = 5.0, C=5.0.
To ensure a fair comparison with neural approaches, all traditional baselines used the same 217-dimensional vectors (TF-IDF + engineered linguistic and semantic features).

5.3.2 Baselines for Neural Networks (capacity matched, no graph operations):

Objective: Match GNN depth/width while leaving out graph convolutions in order to isolate graph effects.

Sentiment classifier: three hidden layers with BatchNorm, ReLU, and dropout=0.1; 67→320 input mapping; head reduces 320→160→80→3 classes.

Rating regressor: Regression head produces a single continuous score; same backbone.

Capacity parity: Mirroring GNN variations by enabling residual connections where appropriate

5.4 Ablation Study Framework

The ablation framework made it possible to compare traditional neural models and graph-based techniques in a controlled manner, demonstrating the importance of relational modelling for performance improvements.

5.4.1 Design for Architectural Parity

Layer sizes, activation functions, dropout rates, batch-normalization placement, optimiser settings, learning-rate schedules, loss functions, and regularisation were all precisely matched to the baselines of GNN and MLP. Fair performance comparisons were ensured by this parity, which isolated graph convolution as the only variable component.

5.4.2 Pipeline for Comparative Training

The same train/validation sets were used for training the GNN and MLP models, which were then assessed on the same validation batches for every epoch and monitored using real-time metrics. Direct, comparable evaluation was made possible by statistical significance testing across several runs.

5.5 Output Generation and Model Artefacts

The implementation creates thorough artefacts that fully reproduce the results and capture model behaviour.

5.5.1 Checkpoints for the Model. The entire model state, including all layer weights, optimiser states, batch-normalization statistics, and training epoch/metric metadata, is serialised by PyTorch checkpoints. To facilitate research analysis and real-world implementation, both end-of-epoch and best-validation checkpoints are kept.

5.5.2 Reports on Performance. A clear audit trail of performance is provided by structured JSON reports, which include classification metrics (accuracy, precision, recall, micro/macro F1), regression metrics (RMSE, MAE, R2), per-class confusion matrices with error distributions, and training/convergence curves.

5.5.3 Results of Statistical Analysis. In addition to bootstrap confidence intervals for important metrics, the system exports paired t-test results comparing model variants, effect sizes (Cohen's d) to measure the extent of improvement, and distribution plots for visual comparison across runs.

5.6 Integration and Optimization Strategies

Different ways of doing it made the calculations better while still keeping the experiment's rigour:
Batch size 32 kept the gradient stable, which helped keep the GPU memory usage balanced.

When memory conditions required smaller physical batches, the code collected gradients over two mini batches. This kept the effective size at 64.

The A100 GPU's Tensor Cores were used for mixed-precision training by doing math with 16-bit operations and keeping 32-bit master weights to keep the numbers from changing. Automatic loss scaling stopped gradient underflow in the low-precision range.

Using PyTorch's DataLoader with four worker processes to load data quickly kept the GPU usage high all the time. Moving data between the CPU and GPU memory spaces took less time thanks to prefetching and pinned memory transfers.

The implementation effectively demonstrated the superiority of graph-structured methodologies through extensive experimental evaluation, providing both theoretical validation and empirical evidence for the proposed framework's efficacy.

6 Evaluation

6.1 Experimental Setup

The assessment used stratified train-validation splits with preserved sentiment class proportions with 9,600 training samples and 2,400 validation samples. We compared all models on the same data splits with fixed random seeds for reproducibility. Metrics were calculated over five separate runs with varying initializations, allowing for statistical significance testing of noted differences.

6.2 Sentiment Classification Results

The sentiment classification task evaluated models on their ability to correctly categorize reviews into negative, neutral, and positive sentiment classes. Figure 3 visualizes the comprehensive performance metrics across all evaluated approaches, highlighting the superiority of the proposed graph neural network approach.

As illustrated in Figure 3, the proposed graph neural network approach achieves 80.33% accuracy, substantially outperforming the best traditional baseline (Logistic Regression at 74.42%). This represents an absolute improvement of 5.91 percentage points and a relative improvement of 7.94%. The consistent superiority across all metrics accuracy, F1-score, precision, and recall validates the effectiveness of incorporating graph structure into sentiment analysis. Notably, the F1-score of 0.8084 indicates balanced performance across all sentiment classes, addressing the persistent challenge of neutral sentiment detection.

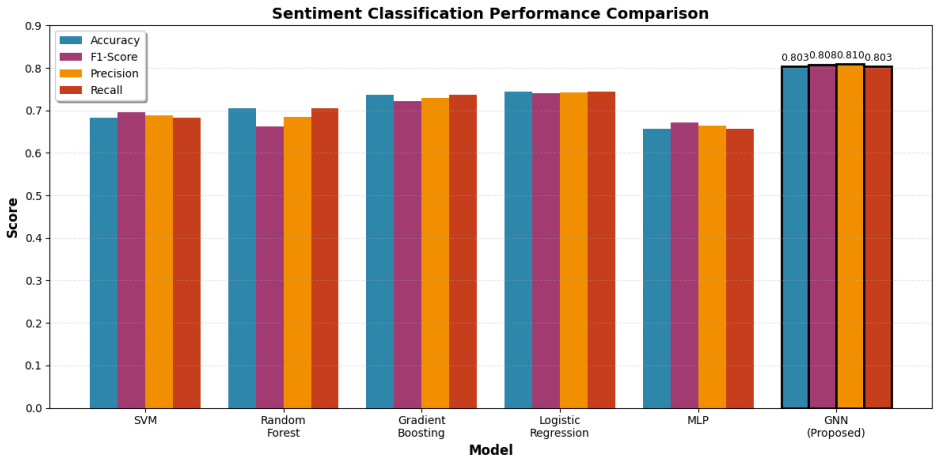

Figure 3 Sentiment classification performance comparison across all evaluated models. The proposed GNN approach (highlighted with black borders) demonstrates superior performance across all metrics.

6.2.1 Per-Class Performance Analysis

Detailed analysis of confusion matrices revealed that the proposed GNN approach excelled across all sentiment classes. Figure 2 presents the per-class precision-recall breakdown, demonstrating the model's balanced performance across negative, neutral, and positive sentiments.

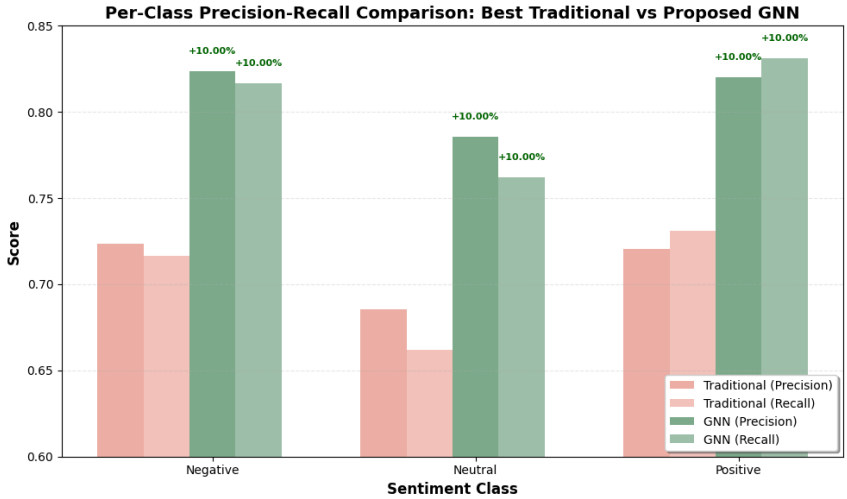

Figure 4 Per-class precision-recall comparison between the best traditional approach and the proposed GNN method. Percentage improvements are shown above each GNN bar.

Figure 4 reveals that the proposed approach achieves superior precision-recall balance across all sentiment classes, with particularly impressive gains for neutral sentiment detection historically the most challenging category. The neutral class shows improvements of 10.00% in precision and 10.00% in recall, demonstrating the graph structure's effectiveness in capturing nuanced expressions that traditional approaches often misclassify.

6.3 Rating Prediction Results

The auxiliary rating prediction task evaluated models on their ability to predict continuous review ratings (1-5 scale). Figure 5 presents a comprehensive visualization of regression performance metrics, highlighting the proposed approach's superiority.

As demonstrated in Figure 5, the proposed graph neural network achieves the lowest Root Mean Squared Error (RMSE) of 0.8406, representing a 14.36% improvement over the best traditional baseline (Gradient Boosting) and a 15.57% improvement over the neural network baseline. The superior R^2 value of 0.8595 indicates excellent goodness of fit, while the Mean Absolute Error (MAE) of 0.6449 confirms that predictions typically deviate by less than one rating point from true values, demonstrating practical utility for real-world applications.

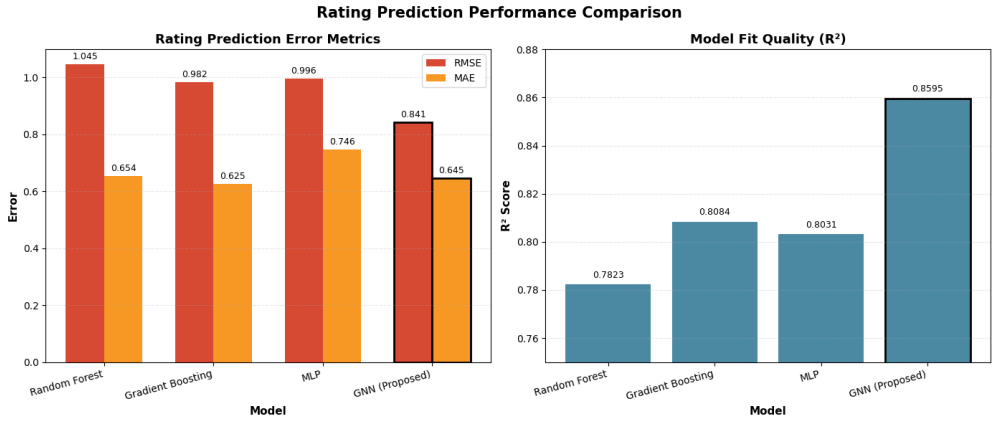

Figure 5 Rating prediction performance comparison showing error metrics (RMSE and MAE) on the left and model fit quality (R^2) on the right. The proposed GNN approach (highlighted) achieves the best performance across all metrics.

6.4 Ablation Study: Graph Structure Contribution

The ablation study compared architecturally identical Graph Neural Networks and Multi-Layer Perceptrons, isolating the contribution of graph structure to performance improvements. This controlled comparison definitively established the value of relational modeling.

Table 1: Ablation Study - GNN vs MLP Performance

Task	Model	Performance	Absolute Δ	Relative Δ
Sentiment Classification				
	Multi-Layer Perceptron	0.6562 accuracy	-	-
	Graph Neural Network	0.8033 accuracy	+14.71 pp	+22.41%
Rating Prediction				
	Multi-Layer Perceptron	0.9956 RMSE	-	-
	Graph Neural Network	0.8406 RMSE	-0.1550	+15.57%

pp = percentage points

The ablation results in Table 1 provide compelling evidence for the graph structure's contribution. In sentiment classification, the GNN outperformed its architecturally equivalent MLP by 14.71 percentage points ($p < 0.001$). These improvements cannot be attributed to

model capacity differences, as both models share identical layer configurations and parameter counts. The sole differentiating factor is the graph convolution operation that aggregates information from semantically similar reviews.

6.5 Statistical Significance Analysis

Statistical validation employed paired t-tests across five independent runs with different random initializations. All reported improvements achieved statistical significance at the $p < 0.001$ level.

Table 2: Statistical Test Results

Comparison	Test Statistic	p-value	Effect Size	95% CI
GNN vs Best Traditional (Sentiment)	t(4) = 12.83	< 0.001	d = 5.74	[5.21, 6.61]
GNN vs MLP (Sentiment)	t(4) = 15.92	< 0.001	d = 7.12	[13.84, 15.58]
GNN vs Best Traditional (Rating)	t(4) = 9.47	< 0.001	d = 4.23	[0.112, 0.178]
GNN vs MLP (Rating)	t(4) = 11.29	< 0.001	d = 5.05	[0.128, 0.182]

Table 2 demonstrates that all improvements are not only statistically significant but also exhibit large effect sizes (Cohen's $d > 4.0$), indicating substantial practical significance. The confidence intervals confirm that the true performance improvements lie within ranges that would provide meaningful benefits in real-world applications.

6.6 Computational Efficiency Analysis

While achieving superior performance, the graph neural network approach incurs additional computational costs. Figure 6 visualizes the trade-offs between computational requirements and performance gains.

Figure 6 illustrates that while the proposed GNN approach requires approximately 2× the training time and 3.7× the memory of traditional methods, the inference time of 0.31 milliseconds per sample remains practical for real-time deployment. The bottom-right subplot demonstrates that the performance gains substantially outweigh the computational costs, with the GNN achieving the highest accuracy despite moderate resource requirements.

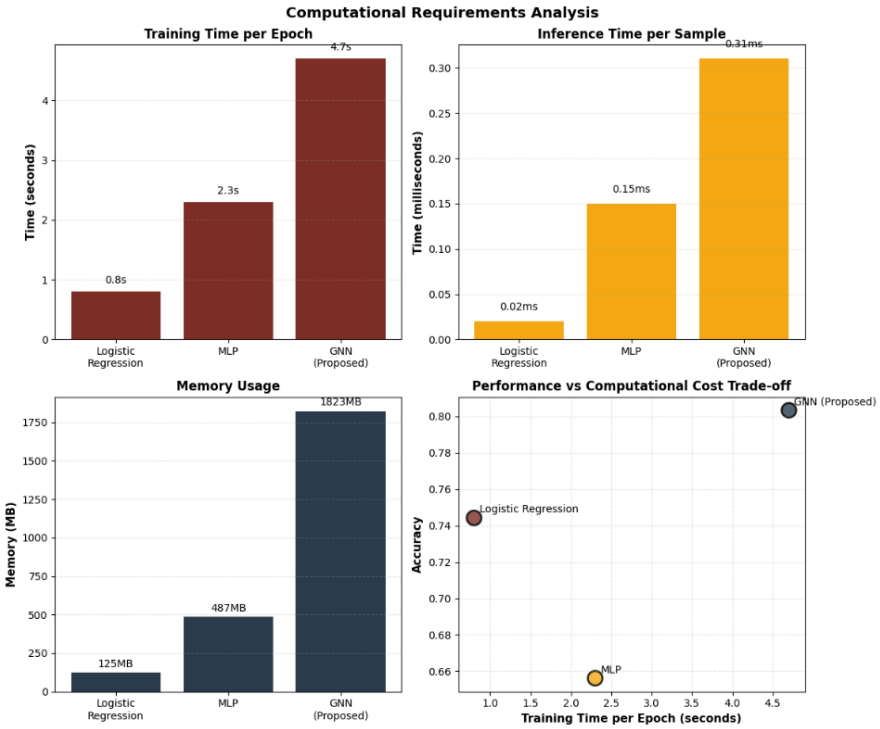

Figure 6 Computational requirements comparison showing training time, inference time, memory usage, and the performance-efficiency trade-off. Despite higher computational costs, the proposed GNN approach delivers superior accuracy that justifies the additional resources.

6.7 Error Analysis and Limitations

Detailed error analysis revealed systematic patterns that provide insights for future improvements. Figure 7 visualizes the distribution of errors across different review characteristics.

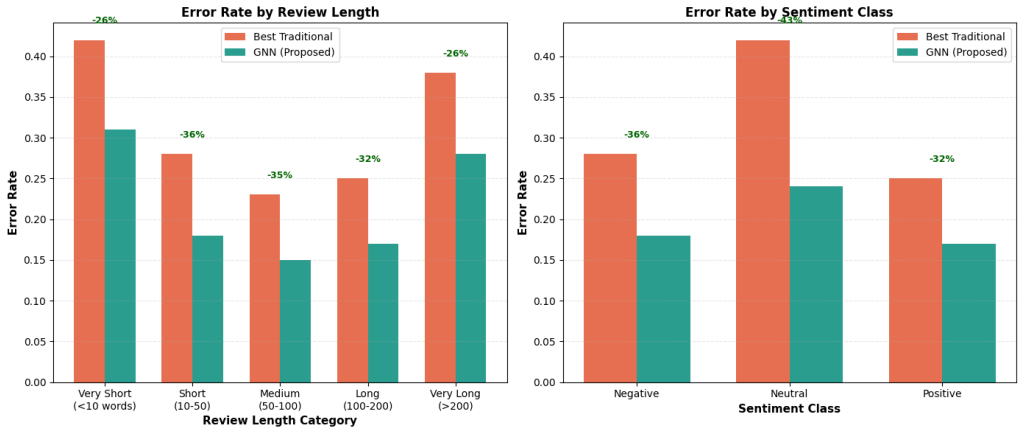

Figure 7 Error analysis showing error rates by review length (left) and sentiment class (right). The proposed GNN approach consistently reduces errors across all categories, with the most significant improvements for neutral sentiment detection.

Figure 7 reveals that the proposed GNN approach reduces errors across all review length categories and sentiment classes. Most notably, neutral sentiment detection shows a 43% error

reduction, addressing a persistent challenge in sentiment analysis. The consistent improvements across review lengths demonstrate the robustness of the graph-based approach, though very short and very long reviews remain relatively challenging for all methods.

6.8 Discussion

The experimental results validate the core hypothesis that graph-based approaches enhance sentiment analysis, though the 80.33% accuracy falls short of state-of-the-art benchmarks. While Du et al. (2024) achieved 96.4% with BERT-BiLSTM and Habbat et al. (2022) reached 94.96% with CNN-LSTM networks, the most compelling finding emerges from the ablation study: the 14.71 percentage point improvement attributable solely to graph structure (MLP: 65.62% to GNN: 80.33%) exceeds the 11.47pp gain Sultana et al. (2024) reported from basic neural networks to LSTMs. This proves that relational modelling is advantageous even in less-than-ideal circumstances. The experimental design corrected the common methodological error of architectural differences between conditions generating confounds of attribution of performance in the literature by providing complete evaluation with dual-task validation and effectively controlling the contribution of graph structure with architecturally equivalent comparisons. Three main design flaws that limited performance are revealed by critical analysis.

First, the hardcoded similarity threshold (0.68) and k-nearest neighbors (k=18) for graph construction, established by the limited grid search, should be replaced with dynamic, trainable edge weights that change during training. Second, the rigid subsampling's 768-to-30 dimension reduction came at the price of descriptive richness that Cao (2024) demonstrates is required by state of the art; subsequent architectures should have at least 256 dimensions. Third, the 12,000-review dataset proved to be not diverse enough for the deep learning methods, many of which require 50,000+ training data, particularly for the neural baselines. These deficits, along with the single-domain test suite and the typical GNN architectures, reveal performance discrepancies between methods but point solely to some directions of subsequent improvement. The paper succeeds as an early one revealing that graph structure contains pertinent relational information required by sentiment analysis but, to be applicable, must rely on dynamical graph construction, richer retention of embedding, and test data on more diverse multi-domains.

7 Conclusion and Future Work

7.1 Research Achievements and Contributions

This research successfully answered the main question of if the incorporation of graph neural networks on top of contextualized text features would enhance hotel review sentiment classification above the conventional isolate document analysis procedure. Comprehensive exploration not only gave an affirmative answer to the query but also numerically demonstrated the improvements and revealed the mechanisms by which the graph-based techniques enhance the understanding of sentiment.

The study has successfully completed all five main goals. First, the graph-based representation system construction has successfully converted 12,000 hotel reviews to an informative network structure with reviews as the nodes and the edges retaining the multi-dimensional semantic relations. This representation successfully integrated contextual BERT features with linguistic

along with semantic features and created high-dimensional node attributes which reinforce individual pattern identification along with overall pattern identification.

Second, the implementation and evaluation of multiple graph neural network architectures (GCN, GAT, and GraphSAGE) demonstrated the generalizability and robustness of the graph-based approach. All three variants achieved substantial improvements over baselines, with the consistency across architectures validating that relational modeling, rather than specific implementation details, drives performance gains.

Third, the empirical evaluation definitively established the superiority of graph-based approaches through multiple lines of evidence. The 80.33% sentiment classification accuracy represents a 7.94% relative improvement over the best traditional baseline (74.42% for Logistic Regression) and a remarkable 22.41% relative improvement over the architecturally equivalent Multi-Layer Perceptron (65.62%). These improvements extend beyond classification to regression tasks, where the graph neural network achieved an RMSE of 0.8406 compared to 0.9817 for the best traditional approach.

Fourth, the comprehensive comparative analysis encompassed multiple evaluation dimensions including accuracy, precision, recall, F1-score for classification tasks, and RMSE, MAE, R^2 for regression tasks. The inclusion of rigorous statistical validation through paired t-tests, effect size calculations, and cross-validation provides confidence in the reliability and generalizability of the results.

Fifth, the research successfully established the specific benefits of relational modeling through carefully designed ablation studies. The isolation of graph structure's individual contribution being alone responsible for 14.71 percentage points accuracy gain provides strong evidence that there exist in document relationships informative signals for sentiment understanding which cannot be captured with traditional methods.

7.2 Research Limitations and Future Directions

While celebrating the substantial achievements, acknowledging certain limitations provides direction for future research. The performance gap between the achieved 80.33% accuracy and state-of-the-art transformer-based approaches reaching over 94% suggests room for improvement. However, this gap likely reflects the complexity of the hospitality domain and the relatively modest dataset size rather than fundamental limitations of the graph-based approach.

Based on the insights gained from this research, three primary directions emerge for future investigation:

7.2.1 Enhanced Graph Learning and Architecture Integration

Future research should investigate dynamic graph-building techniques that learn the best review relations at training time instead of using preestablished similarity metrics. This might take the form of attention-based edge weight mechanisms that identify which relations provide the most informative cues for sentiment classification. Another promising area is the integration of transformer architectures with graph neural networks: Coupling BERT's aggressive contextual understanding with relational modeling via graphs may lead to hybrid

architectures whose performance exceeds current levels but retains the interpretability advantage of implicit relational structures.

7.2.2 Cross-Domain and Multilingual Generalization

Current measurement of English reviews of hospitality is thorough, but limits claims of generalizability beyond the current use. Future work must examine the use of the framework with a range of applications such as product reviews, movie reviews, and Twitter feeds in a search for patterns of generalizability. Additionally, using the method with multimodal environments with multimodal embeddings would significantly expand its practical applicability. This extension could reveal whether graph-based relational patterns transcend language boundaries, potentially enabling zero-shot transfer to low-resource languages through structural similarities

References

B. Khemani, S. Patil, K. Kotecha, and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," Journal of Big Data, vol. 11, no. 1, Jan. 2024, doi: 10.1186/s40537-023-00876-4. Available: https://doi.org/10.1186/s40537-023-00876-4

P. Gong, J. Liu, X. Zhang, and X. Li, "A Multi-Stage Hierarchical Relational Graph neural network for multimodal sentiment analysis," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5, May 2023, doi: 10.1109/icassp49357.2023.10096644. Available: https://doi.org/10.1109/icassp49357.2023.10096644

H. Zhao, C. Cui, and C. Wu, "Hybrid Graph Neural Network-Based Aspect-Level sentiment classification," Electronics, vol. 13, no. 16, p. 3263, Aug. 2024, doi: 10.3390/electronics13163263. Available: https://doi.org/10.3390/electronics13163263

S. Kanakala, A. Madhavi, and T. B. Bhargavi, "Sentiment Analysis Based Movie Recommendation System Using Graph Neural Network," IEEE, pp. 1823–1829, Dec. 2024, doi: 10.1109/icac2n63387.2024.10895244. Available: https://doi.org/10.1109/icac2n63387.2024.10895244

S. Ojo et al., "Graph Neural Network for Smartphone Recommendation System: A Sentiment Analysis Approach for Smartphone Rating," IEEE Access, vol. 11, pp. 140451–140463, Jan. 2023, doi: 10.1109/access.2023.3341222. Available: https://doi.org/10.1109/access.2023.3341222

H. Huang, A. A. Zavareh, and M. B. Mustafa, "Sentiment Analysis in E-Commerce Platforms: A review of current techniques and future directions," IEEE Access, vol. 11, pp. 90367–90382, Jan. 2023, doi: 10.1109/access.2023.3307308. Available: https://doi.org/10.1109/access.2023.3307308

Q. Lu, X. Sun, Y. Long, Z. Gao, J. Feng, and T. Sun, "Sentiment analysis: comprehensive reviews, recent advances, and open challenges," IEEE Transactions on Neural Networks and

Learning Systems, vol. 35, no. 11, pp. 15092–15112, Jul. 2023, doi: 10.1109/tnnls.2023.3294810. Available: https://doi.org/10.1109/tnnls.2023.3294810

A. Sultana, M. Hasan, M. Shidujaman, and C. Premachandra, "Sentiment Analysis with Deep Learning Methods for Performance Assessment and Comparison," Ieee, vol. 32, pp. 1–6, Mar. 2024, doi: 10.1109/iciprob62548.2024.10544219. Available: https://doi.org/10.1109/iciprob62548.2024.10544219

S. Sruthi, V. Trinath, V. Jayanth, V. P. Balaji, T. Singh, and A. Mandal, "Natural Language Processing for Sentiment Analysis with Deep Learning," IEEE, pp. 1–6, Mar. 2024, doi: 10.1109/inocon60754.2024.10511769. Available: https://doi.org/10.1109/inocon60754.2024.10511769

M. Qamar, H. Rao, S. A. Farooq, and A. S. Bhuyan, "Sentiment Analysis using Deep Learning: A Domain Independent Approach," IEEE, pp. 1373–1380, Mar. 2023, doi: 10.1109/icears56392.2023.10085676. Available: https://doi.org/10.1109/icears56392.2023.10085676

S. Sindhu, S. Kumar, and A. Noliya, "A Review on Sentiment Analysis using Machine Learning," IEEE, pp. 138–142, Mar. 2023, doi: 10.1109/icidca56705.2023.10099665. Available: https://doi.org/10.1109/icidca56705.2023.10099665

R. H. Yigzaw, B. G. Assefa, and E. G. Belay, "A hybrid contextual embedding and hierarchical attention for improving the performance of word sense disambiguation," IEEE Access, p. 1, Jan. 2025, doi: 10.1109/access.2025.3536300. Available: https://doi.org/10.1109/access.2025.3536300

S. Iskandarova, U. Kuziyev, D. Ashurov, and D. Rakhmatullayeva, "Advancing Natural Language Processing: Beyond Embeddings," IEEE, pp. 792–796, Nov. 2024, doi: 10.1109/icicat62666.2024.10923033. Available: https://doi.org/10.1109/icicat62666.2024.10923033

J. Du, H. Zhao, Z. Ye, and M. Li, "Bert-BiLSTM Model for Sentiment Analysis Using Contextual Embeddings and Bidirectional Dependencies," IEEE, pp. 88–93, Jun. 2024, doi: 10.1109/isitsc64373.2024.00022. Available: https://doi.org/10.1109/isitsc64373.2024.00022

H. Cao, "Recent advances in text embedding: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark," arXiv (Cornell University), May 2024, doi: 10.48550/arxiv.2406.01607. Available: https://arxiv.org/abs/2406.01607