Responses to the Reviews of the Reviewers

January 3, 2026

We thank the Associate Editor and the reviewers for their careful reading of our paper “Graph Enhanced Transformer for Semi-Supervised Duplicate Bug Report Detection" and their constructive Reviews. We also thank the reviewers for pointing out our novel contributions. In the revised version of the paper (and in the response document below), we have addressed each of the issues raised by the reviewers. Below, we provide a point by point response to the Reviews of the reviewers and how they are handled in the revised manuscript. Since we have answered each Review separately, there are some overlapping answers. Significant changes in the revised paper are also marked in blue.

We thank the editor and reviewers for their thorough and constructive feedbacks. We have carefully addressed all the Reviews. In the revised document, we have made the following main improvements to the manuscript:

1. **Expanded Future Work and Limitations:** Addressed reviewer suggestions regarding PCA sensitivity, label-scarcity experiments, and hard negative mining by providing detailed discussions in the expanded “Future Work" and “Threats to Validity" sections.

2. **Performance and Reproducibility:** Contextualized the competitive performance of our semi-supervised framework and fully updated the GitHub replication package to ensure complete transparency and reproducibility.

3. **Quality Control and Formatting:** Conducted a comprehensive manual review of the manuscript to correct grammatical errors, duplicated phrases, and terminological inconsistencies pointed out by the reviewers. Substantially restructured the “Problem Definition and Motivation" section to eliminate redundancies and include concrete examples of LLM failures and formal DBRD definitions.

We hope these revisions address all reviewer concerns and significantly improve the quality and clarity of the manuscript.

Reviewer 1

Overall Recommendation: Weak Accept

Review 1.1

In Section 5.4 (lines 955–970), the paper discusses the training process by only mentioning starting and ending loss values. It would be more helpful to reference the figures before starting the discussion or include loss curves at the same page to improve reading flow.

Response: We thank the reviewer for this valuable suggestion. We agree that explicitly referencing the convergence figures improves both clarity and flow. We have revised the Training Dynamics Analysis section to include explicit references to Figures 3 and 4 at the beginning of the discussion.

Revisions in Manuscript:

↪ **Page 9, Training Dynamics Analysis:**
"The detailed convergence plots are shown in Fig. 3 and 4 for Eclipse and Thunderbird datasets, respectively."

Review 1.2

Potential improvements include experimenting with pretrained embeddings or learned node features instead of one-hot encodings, testing different PCA dimensionalities, and conducting ablations for hyperparameters such as projection dimension D, fusion weight λ, margin m, and training epochs.

Response: We thank the reviewer for these thoughtful suggestions. We fully acknowledge the value of these ablation studies and sensitivity analyses. Due to computational resource and time constraints, we were unable to conduct comprehensive hyperparameter exploration in this revision. However, we have now explicitly discussed these directions in the expanded Future Work section.

Revisions in Manuscript:

↪ **Page 10, Hyperparameter Sensitivity and Ablation Studies:**
"Several hyperparameters in our framework were fixed based on preliminary experiments without exhaustive ablation studies. The PCA dimensionality (d=10), projection dimension (D=128), fusion weight (λ), margin (m) in the loss function, and number of training epochs all warrant systematic sensitivity analysis. In particular, the aggressive dimensionality reduction from 768 to 10 dimensions via PCA was chosen empirically to improve discrimination but may introduce information loss. Future work should explore different values of d (e.g., 5, 20, 50, 100) and characterize the trade-off between noise reduction and information preservation across different datasets. Such ablation studies would provide data-driven justification for hyperparameter choices and reveal the robustness of the proposed framework."

Review 1.3

Grammatical issues: GCN used before definition, keyword capitalization, "ie." format, duplicated "for example for example", missing commas, and "token id's" formatting.

Response: We thank the reviewer for bringing these issues to our attention. We have systematically corrected all mentioned errors throughout the manuscript.

Revisions in Manuscript:

↪ **Page 1, Abstract:**
"Graph Convolutional Network (GCN)"

↪ ***Page 2, Introduction:***
"for example, 20% of reports in Eclipse and 30% in Firefox were marked as duplicate"

↪ ***Page 4, Proposed Method:***
"i.e., the negative pair sampling"

Review 1.4

The replication package is missing key files and the training notebook shows interrupted execution.

Response: We thank the reviewer for this observation. We have updated our GitHub repository to include all necessary files for complete reproducibility and clarified the repository pointer in the manuscript. We would like to point out that the repository README has a Google Drive link, containing preprocessing codes and checkpoints. These files are not included in the main GitHub repository to keep the size small and focus on the main code.

Revisions in Manuscript:

↪ **Page 8, Implementation Details:**
"All source code and the full reproducibility package, including all necessary data files and scripts, can be found in the huseyin-karaca/graph-enhanced-dbd GitHub repository 🔗."

Reviewer 2

Overall Recommendation: Weak Reject

Review 2.1

The paper claims to address "label-scarce" environments but uses full datasets. Experiments with reduced training set sizes (5% or 10% of labeled data) are needed.

Response: We thank the reviewer for this important observation. We first want to clarify that the current experiments use full training datasets for only graph construction, not for transformer training. We acknowledge that our current experiments do not directly validate performance under extreme label scarcity (approximately 80% for now). While the graph-based framework is designed to leverage unlabeled data, we agree that experiments with systematically reduced label budgets would provide stronger evidence. Due to time and computational constraints, we could not complete these experiments for this revision. We have added this explicitly to the Future Work section.

Revisions in Manuscript:

↪ **Page 10, Label-Scarce Validation and Inference Latency:**
"An important remark is that the current experiments use full training datasets for only graph construction, not for transformer training. A key motivation for graph-based semi-supervised learning is its potential effectiveness under label-scarce conditions. Currently, we use approximately 80% of the training data for transformer training. Future work should include experiments with reduced training set sizes (e.g., 5% or 10% of labeled data) to empirically validate the claim that graph propagation provides concrete benefits when annotations are limited. Additionally, while we have characterized training-time overhead, future work should include detailed inference latency measurements (e.g., milliseconds per query) comparing the proposed method against baselines, providing quantitative evidence for the inference scalability claims made in RQ2."

Review 2.2

Table 4 reports only training time overhead. Inference latency metrics must be provided to support scalability claims.

Response: We thank the reviewer for this valuable point. We agree that inference latency measurements would strengthen RQ2. We have added quantitative inference benchmarking as future work.

Revisions in Manuscript:

↪ **Page 10, Label-Scarce Validation and Inference Latency:**
"An important remark is that the current experiments use full training datasets for only graph construction, not for transformer training. A key motivation for graph-based semi-supervised learning is its potential effectiveness under label-scarce conditions. Currently, we use approximately 80% of the training data for transformer training. Future work should include experiments with reduced training set sizes (e.g., 5% or 10% of labeled data) to empirically validate the claim that graph propagation provides concrete benefits when annotations are limited. Additionally, while we have characterized training-time overhead, future work should include detailed inference latency measurements (e.g., milliseconds per query) comparing the proposed method against baselines, providing quantitative evidence for the inference scalability claims made in RQ2."

Review 2.3

The PCA dimensionality reduction from 768 to d=10 appears arbitrary. Sensitivity analysis should be conducted.

Response: We thank the reviewer for raising this concern. We acknowledge that d=10 was empirically motivated but not systematically validated. Due to computational constraints, we have expanded the Future Work section to explicitly highlight the need for PCA dimensionality sensitivity analysis.

Revisions in Manuscript:

↪ **Page 10, Hyperparameter Sensitivity and Ablation Studies:**
"Several hyperparameters in our framework were fixed based on preliminary experiments without exhaustive ablation studies. The PCA dimensionality (d=10), projection dimension (D=128), fusion weight (λ), margin (m) in the loss function, and number of training epochs all warrant systematic sensitivity analysis. In particular, the aggressive dimensionality reduction from 768 to 10 dimensions via PCA was chosen empirically to improve discrimination but may introduce information loss. Future work should explore different values of d (e.g., 5, 20, 50, 100) and characterize the trade-off between noise reduction and information preservation across different datasets. Such ablation studies would provide data-driven justification for hyperparameter choices and reveal the robustness of the proposed framework."

Review 2.4

Table 5 shows the method provides no concrete benefit over baselines; results are identical or slightly worse.

Response: We thank the reviewer for this observation. We respectfully emphasize that our goal is to demonstrate that a novel graph-enhanced semi-supervised framework can achieve *competitive* performance while explicitly leveraging unlabeled data. Achieving F1 scores that match or closely approximate state-of-the-art models validates the architectural concept. We have added a paragraph emphasizing this perspective.

Revisions in Manuscript:

↪ **Page 9, Competitive Performance Evaluation (RQ3):**
"It is important to emphasize that the goal of this work is not merely to achieve marginal numerical improvements over existing baselines, but rather to demonstrate that a graph-enhanced semi-supervised framework can reach competitive performance while explicitly leveraging unlabeled data during training. The fact that our approach matches or closely approximates the performance of heavily-optimized, fully-supervised transformer models while incorporating structural information from unlabeled reports represents a meaningful contribution. This validates the architectural concept and establishes that graph-based semi-supervised learning is a viable path forward for duplicate bug report detection in label-scarce settings, even if the current instantiation does not universally outperform all baselines."

Review 2.5

While the GNN application shows originality, similar hybrid architectures exist in other NLP domains, limiting foundational novelty.

Response: We thank the reviewer for this fair assessment. We acknowledge that hybrid architectures have been explored elsewhere. However, to the best of our knowledge, this work represents an early exploration of graph-based semi-supervised learning specifically for duplicate bug report detection, with distinct domain characteristics and design choices.

Revisions in Manuscript: No change was required in the manuscript.

Review 2.6

The paper is well-written. However, "for example" is duplicated in Line 120.

Response: We thank the reviewer for the positive feedback. We have corrected the duplicated phrase.

Revisions in Manuscript:

↪ **Page 2, Introduction:**
"for example, 20% of reports in Eclipse and 30% in Firefox were marked as duplicate"

Reviewer 3

Overall Recommendation: Weak Reject

Review 3.1

Graph edges use only titles. Ignoring descriptions may discard important relationships.

Response: We thank the reviewer for this insightful observation. We fully agree that descriptions could yield richer structures. Our choice was motivated by computational efficiency. We have expanded Future Work to discuss incorporating descriptions.

Revisions in Manuscript:

↪ **Page 11, Incorporating Descriptions in Graph Construction:**
"In this work, graph edges are constructed based solely on semantic similarity of bug report titles. While titles provide concise summaries, they are often brief, vague, and incomplete compared to full descriptions. Incorporating description text when computing semantic similarity for edge formation could yield substantially richer relational structures. However, this introduces computational challenges (longer sequences, higher memory requirements) and potential noise (descriptions may contain less relevant information). Future work should explore hybrid approaches that weight title and description similarity, or use multi-view graph construction that creates separate edge types based on different text fields."

Review 3.2

PCA reduction from 768 to 10 dimensions is very aggressive and may not generalize.

Response: We thank the reviewer for this valid concern. We acknowledge the aggressive reduction and its generalization risks. We have expanded Future Work to highlight the need for PCA dimensionality ablation studies.

Revisions in Manuscript:

↪ **Page 10, Hyperparameter Sensitivity and Ablation Studies:**
"Several hyperparameters in our framework were fixed based on preliminary experiments without exhaustive ablation studies. The PCA dimensionality (d=10), projection dimension (D=128), fusion weight (λ), margin (m) in the loss function, and number of training epochs all warrant systematic sensitivity analysis. In particular, the aggressive dimensionality reduction from 768 to 10 dimensions via PCA was chosen empirically to improve discrimination but may introduce information loss. Future work should explore different values of d (e.g., 5, 20, 50, 100) and characterize the trade-off between noise reduction and information preservation across different datasets. Such ablation studies would provide data-driven justification for hyperparameter choices and reveal the robustness of the proposed framework."

Review 3.3

Negative sampling may create mostly easy negatives. Hard negative mining could improve decision boundaries.

Response: We thank the reviewer for this point. While our approach combines two strategies to ensure balance, we acknowledge that explicit hard negative mining could further sharpen boundaries. We have added this to Future Work.

Revisions in Manuscript:

↪ **Page 11, Hard Negative Mining and Advanced Sampling:**
"The current negative sampling strategy combines anchor-based pairing with random sampling across duplicate groups. While this ensures balanced representation of positive and negative examples, it may over-represent easy negatives—report pairs that are clearly dissimilar. Hard negative mining, which focuses on difficult non-duplicates (reports that appear similar but describe different bugs), could improve the model's ability to learn fine-grained decision boundaries. Future work should explore curriculum-based training strategies that progressively introduce harder negatives, or employ similarity-based negative sampling to deliberately select challenging negative pairs."

Review 3.4

Maintaining the full graph in memory may become difficult as repositories grow.

Response: We thank the reviewer for this practical concern. For even larger repositories, we discuss strategies in the expanded Future Work section.

Revisions in Manuscript:

↪ **Page 11, Distributed Graph Construction and Storage:**
"For extremely large bug repositories (e.g., hundreds of thousands or millions of reports), maintaining the full graph in memory on a single GPU becomes impractical. Future work should investigate distributed computing strategies for both graph construction and GNN training. Techniques such as graph partitioning across multiple GPUs or machines, distributed message passing frameworks (e.g., DistDGL, PyTorch Geometric distributed), and out-of-core graph storage could enable scaling to industrial-scale repositories. Additionally, approximate methods such as graph sampling or mini-batch GNN training on subgraphs could reduce memory footprint while maintaining representational quality."

Review 3.5

Evaluation limited to Eclipse and Thunderbird constrains generalization claims.

Response: We thank the reviewer for this valid observation. We acknowledge this limitation and have listed evaluation on diverse datasets as future work.

Revisions in Manuscript:

↪ **Page 11, Evaluation on Diverse Datasets:**
"Our evaluation is limited to two benchmark datasets from large, open-source projects (Eclipse and Thunderbird). These repositories may exhibit similar reporting cultures and technical domains. To assess the generalizability of the proposed framework, future work should evaluate performance on a more diverse set of repositories, including smaller projects, proprietary software systems, and domains with different bug reporting guidelines (e.g., mobile applications, embedded systems, web services). Cross-domain evaluation would reveal whether the graph-enhanced approach is robust to variations in vocabulary, reporting style, and duplicate patterns."

Review 3.6

Typos and awkward phrases, such as "for example for example".

Response: We thank the reviewer. We have corrected the duplicated phrase and reviewed the manuscript for similar issues.

Revisions in Manuscript:

↪ **Page 2, Introduction:**
"for example, 20% of reports in Eclipse and 30% in Firefox were marked as duplicate"

Review 3.7

Results don't beat baselines; lack of cross-validation or statistical tests makes small differences unreliable.

Response: We thank the reviewer. We acknowledge both concerns: (1) our goal is competitive performance with unlabeled data leverage, and (2) lack of statistical rigor is a limitation. We have acknowledged this in Threats to Validity.

Revisions in Manuscript:

↪ **Page 9, Competitive Performance Evaluation (RQ3):**
"It is important to emphasize that the goal of this work is not merely to achieve marginal numerical improvements over existing baselines, but rather to demonstrate that a graph-enhanced semi-supervised framework can reach competitive performance while explicitly leveraging unlabeled data during training. The fact that our approach matches or closely approximates the performance of heavily-optimized, fully-supervised transformer models while incorporating structural information from unlabeled reports represents a meaningful contribution. This validates the architectural concept and establishes that graph-based semi-supervised learning is a viable path forward for duplicate bug report detection in label-scarce settings, even if the current instantiation does not universally outperform all baselines."

Review 3.8

Overlap between Introduction, Problem Description, and Related Work sections.

Response: We thank the reviewer for this structural feedback. We have substantially revised Section 2 to focus on problem definition and motivation, reducing redundancy.

Revisions in Manuscript:

↪ **Page 2, Formal Problem Definition:**
"The core challenge in Duplicate Bug Report Detection (DBRD) is to automatically identify whether two bug reports describe the same underlying software defect. Formally, given a bug repository $\mathcal{R} = \{r_1, r_2, ..., r_n\}$ where each report r_i contains textual fields (title, description) and metadata (timestamp, reporter, component), the task is to learn a function $f : \mathcal{R} \times \mathcal{R} \to \{0, 1\}$ that predicts whether a pair of reports (r_i, r_j) are duplicates. This is inherently difficult due to lexical variations, incomplete descriptions, domain-specific terminology, and reporting style differences across users."

↪ ***Page 2, Limitations of Existing LLM-Based Approaches:***
"Transformer-based language models such as BERT, RoBERTa, and their domain-adapted variants have demonstrated strong performance in semantic text matching tasks. However, these models face fundamental limitations in the DBRD setting. Consider a concrete scenario: a repository contains 20,000 bug reports with only 500 confirmed duplicate pairs (1,000 labeled reports). A pairwise BERT-based classifier would be trained on these 500 positive pairs plus an equal number of sampled negative pairs, effectively utilizing only 1,000 reports while completely disregarding the remaining 19,000 unlabeled reports."

↪ ***Page 3, Illustrative Example: LLM Failure Under Data Scarcity:***
"To illustrate the practical limitation, consider two bug reports from an Eclipse repository: Report A (ID 12345): "NullPointerException in editor when opening XML file with invalid schema reference" Report B (ID 67890): "NPE thrown during XML editor initialization with broken schema link." These reports clearly describe the same bug using different terminology ("NullPointerException" vs. "NPE", "opening" vs. "initialization", "invalid reference" vs. "broken link"). A well-trained transformer model with sufficient labeled examples from the XML editor domain would likely identify this pair as duplicates through semantic similarity. However, suppose the training set contains only 100 labeled duplicate pairs, none of which involve the XML editor component or schema-related issues. The transformer model, trained exclusively on these 100 pairs, would lack the domain-specific context to recognize the semantic equivalence between Reports A and B. Meanwhile, the repository contains 500 other unlabeled reports related to XML editing, including variations of similar terminology and error patterns. A purely supervised LLM-based approach cannot leverage these 500 reports during training, as they lack explicit duplicate labels."

↪ ***Page 3, Motivation for Graph-Based Semi-Supervised Learning:***
"Graph neural networks provide a natural solution to overcome these limitations. Unlike pairwise supervised learning, GNNs operate on relational neighborhoods and propagate information through message passing across connected nodes without requiring explicit labels for every connection. This enables us to construct a unified graph representation where all available bug reports—both labeled and unlabeled—participate in the learning process."

Reviewer 4

Overall Recommendation: Weak Accept

Review 4.1

Some parts of the paper seem to be organized inappropriately; especially the Problem Description and Motivation section reads like a summary of the methodology.

Response: We thank the reviewer for this structural critique. We have completely restructured Section 2 to clearly separate problem definition from methodology. It now has the following subsections to improve readability and semantic clarity:

Revisions in Manuscript:

↪ **Page 2, Formal Problem Definition:**
"The core challenge in Duplicate Bug Report Detection (DBRD) is to automatically identify whether two bug reports describe the same underlying software defect. Formally, given a bug repository $\mathcal{R} = \{r_1, r_2, ..., r_n\}$ where each report r_i contains textual fields (title, description) and metadata (timestamp, reporter, component), the task is to learn a function $f : \mathcal{R} \times \mathcal{R} \rightarrow \{0, 1\}$ that predicts whether a pair of reports (r_i, r_j) are duplicates. This is inherently difficult due to lexical variations, incomplete descriptions, domain-specific terminology, and reporting style differences across users."

↪ ***Page 2, Limitations of Existing LLM-Based Approaches:***
"Transformer-based language models such as BERT, RoBERTa, and their domain-adapted variants have demonstrated strong performance in semantic text matching tasks. However, these models face fundamental limitations in the DBRD setting. Consider a concrete scenario: a repository contains 20,000 bug reports with only 500 confirmed duplicate pairs (1,000 labeled reports). A pairwise BERT-based classifier would be trained on these 500 positive pairs plus an equal number of sampled negative pairs, effectively utilizing only 1,000 reports while completely disregarding the remaining 19,000 unlabeled reports."

↪ ***Page 3, Illustrative Example: LLM Failure Under Data Scarcity:***
"To illustrate the practical limitation, consider two bug reports from an Eclipse repository: Report A (ID 12345): "NullPointerException in editor when opening XML file with invalid schema reference" Report B (ID 67890): "NPE thrown during XML editor initialization with broken schema link." These reports clearly describe the same bug using different terminology ("NullPointerException" vs. "NPE", "opening" vs. "initialization", "invalid reference" vs. "broken link"). A well-trained transformer model with sufficient labeled examples from the XML editor domain would likely identify this pair as duplicates through semantic similarity. However, suppose the training set contains only 100 labeled duplicate pairs, none of which involve the XML editor component or schema-related issues. The transformer model, trained exclusively on these 100 pairs, would lack the domain-specific context to recognize the semantic equivalence between Reports A and B. Meanwhile, the repository contains 500 other unlabeled reports related to XML editing, including variations of similar terminology and error patterns. A purely supervised LLM-based approach cannot leverage these 500 reports during training, as they lack explicit duplicate labels."

↪ ***Page 3, Motivation for Graph-Based Semi-Supervised Learning:***
"Graph neural networks provide a natural solution to overcome these limitations. Unlike pairwise supervised learning, GNNs operate on relational neighborhoods and propagate information through message passing across connected nodes without requiring explicit labels for every connection. This enables us to construct a unified graph representation where all available bug reports—both labeled and unlabeled—participate in the learning process."

Review 4.2

Graph construction ignores descriptions, which may forgo significant data.

Response: We thank the reviewer. We acknowledge this limitation and have listed incorporating descriptions as future work.

Revisions in Manuscript:

↪ **Page 11, Incorporating Descriptions in Graph Construction:**
"In this work, graph edges are constructed based solely on semantic similarity of bug report titles. While titles provide concise summaries, they are often brief, vague, and incomplete compared to full descriptions. Incorporating description text when computing semantic similarity for edge formation could yield substantially richer relational structures. However, this introduces computational challenges (longer sequences, higher memory requirements) and potential noise (descriptions may contain less relevant information). Future work should explore hybrid approaches that weight title and description similarity, or use multi-view graph construction that creates separate edge types based on different text fields."

Review 4.3

Various formatting and terminological issues throughout the manuscript.

Response: We thank the reviewer for meticulous attention to detail. We have systematically corrected all mentioned issues. Some examples of the changes are listed below.

Revisions in Manuscript:

↪ **Page 1, Abstract:**
"Graph Convolutional Network (GCN)"

↪ ***Page 1, Introduction:***
"bug repositories"

↪ ***Page 3, Machine Learning Approaches:***
"where N denotes the total number of bug reports in the repository"

↪ ***Page 4, Proposed Method:***
"i.e., the negative pair sampling"

↪ ***Page 5, Embedding and Graph Construction:***
"Graph Construction,"

↪ ***Page 7, Implementation Details:***
"early stopping based on validation performance; specifically, training is stopped if validation loss does not improve for 3 consecutive epochs (patience=3)"