ISSN 1846-6168 (Print), ISSN 1848-5588 (Online)
https://doi.org/10.31803/tg-20250325111250

Original scientific paper
Received: 2025-03-25, Accepted: 2025-09-02

Hierarchical Ensemble Learning for Automatic Classification of Construction Work Codes

Yeong-Chae Yun, Seok-Heon Yun*

Abstract: The Work Classification Code (WCC) serves as a critical tool for standardizing Bill of Quantities (BOQ) in construction projects, ensuring reliable cost estimation and effective expense management. However, current BOQ–WCC matching processes rely heavily on subjective judgment, resulting in inconsistencies and limited standardization. To address this issue, this study constructed a dataset of 8,021 valid items collected from six school construction BOQs commissioned by the Korea Public Procurement Service (PPS), and developed an automated matching model using hierarchical ensemble learning. The proposed approach reflects the five-level hierarchical structure of the WCC (Levels 1–5) and applies different ensemble methods according to classification complexity: Bagging at Levels 1–2, Boosting at Levels 3–4, and Weighted Voting at Level 5. A baseline comparison with a single LSTM model was also conducted, confirming that ensemble methods consistently outperformed the base model, particularly at intermediate levels where classification complexity is higher. Experimental results demonstrated that Bagging provided stable improvements at upper levels, Boosting achieved substantial gains at intermediate levels, while Weighted Voting offered limited benefits at the lowest level. Despite these improvements, overall performance declined as class granularity increased, and Level 5 classification showed severe degradation due to extreme data sparsity. This study confirms the feasibility of applying hierarchical ensemble learning for automated WCC matching and highlights its potential for improving BOQ standardization in real-world projects. The findings suggest that automatic WCC assignment can support reliable cost reviews and enable retroactive coding of BOQs where codes were previously absent, thereby enhancing efficiency in design-phase verification systems. Future work will focus on addressing data sparsity through large-scale BOQ collection, incorporating advanced pre-trained language models such as BERT, refining the hierarchical code structure through data integration, and validating model applicability in public procurement and contractor systems.

Keywords: Automatic Code Matching; Bill of Quantities; Construction Work Classification Code; Ensemble Learning; Hierarchical Classification; Machine Learning

1 INTRODUCTION

In construction projects, the Bill of Quantities (BOQ) is the key document for cost estimation and contract administration. The BOQ systematically lists item descriptions, specifications, units, quantities, unit prices, and the total construction cost, and it is prepared on the basis of design drawings and specifications; thus, it serves as the benchmark for deriving construction costs [1, 2]. This enables cost review and comparison, and—particularly in public projects—clarifies contract terms and enhances the reliability of cost management. To distinguish trades and items, BOQs are accompanied by construction codes, which play an important role in item identification and cost organization. However, the construction codes currently in use vary across agencies and projects, standardization is insufficient, and even identical trades may be expressed differently; moreover, subjective judgment by the preparer is easily introduced [1, 3]. As a result, inter-project cost comparisons become difficult and data consistency cannot be ensured.

To address these issues, Korea's Public Procurement Service provides the Work Classification Code (WCC), which guides the preparation of standardized BOQs. Through a 12-digit hierarchical structure, the WCC systematically classifies trades, standardizes BOQ composition, and facilitates cost comparison and review across projects [4]. Nevertheless, limited practitioner understanding of the WCC's structure and application often leads to the continued use of legacy practices, constraining its uptake in practice.

The purpose of this study is to develop an algorithm for automatically matching BOQ items with the standard WCC. To this end, a hierarchical classification model based on ensemble learning was designed, incorporating Bagging (Bootstrap Aggregating), Boosting, and Weighted Voting techniques at different levels to evaluate performance. The aim is to support BOQ standardization, enhance the reliability of cost review, and reduce the complexity of practitioners' work processes.

The remainder of this paper is structured as follows. First, prior research related to the WCC system is reviewed, and the structure and hierarchical characteristics of the WCC are described. The methodology and hierarchical ensemble model are then presented, followed by an analysis of the experimental results. Finally, the study's contributions and directions for future research are discussed.

2 LITERATURE REVIEW

To systematically review construction costs and manage project expenses, the Korea Public Procurement Service introduced the Construction Work Classification Code (WCC). WCC adopts a hierarchical classification structure, enabling precise categorization of construction work items and ensuring consistency in bill of quantities (BOQ) preparation [4]. According to Lee et al. [4], the application of WCC enhances cost review, supports comparative analysis, and provides a more systematic framework for managing cost variations across trades. However, challenges remain in effectively adopting WCC in practice [5].

One key reason for this limited adoption is the complexity of the coding structure. The WCC consists of a 12-digit hierarchical system, with distinct classification rules at each level. Practitioners often lack sufficient understanding of these rules and continue to prepare BOQs in traditional ways, leaving room for subjective judgment in code selection. Moreover, identical work items may be represented differently across projects, leading to inconsistencies in code application. Yun [6] also emphasized that the absence of a fully standardized coding system has resulted in institutions and designers relying on different construction codes, undermining interoperability.

This standardization challenge is not unique to Korea. In North America, databases based on MasterFormat and

OmniClass have been established, with Guven et al. [7] highlighting the necessity of systematic classification to track and manage construction resources. In the UK, Pupeikis et al. [8] (2022) compared Uniclass 2015 and CCI, noting both the potential of hierarchical classification systems in BIM environments and the practical challenges of implementation. Similarly, Danusevics et al. [9] (2022) conducted a comparative analysis of more than 20 national classification systems, emphasizing that while standardized coding offers clear advantages, its adoption and challenges vary across countries, underscoring the importance of international harmonization. In Spain, policy-level initiatives have begun requiring the integration of BIM-based classification systems and a Common Data Environment (CDE) in public procurement, signaling institutional efforts to enhance consistency in BOQ preparation and data management [10]. Collectively, these international cases demonstrate that the classification and standardization of construction work items are universal challenges across the construction industry.

At the same time, the BOQ consists of textual data representing diverse work items, making effective text classification essential for structuring specifications, organizing documents, and supporting cost estimation [11]. Given the variety of expressions that may be used in BOQ descriptions, recent studies have increasingly applied natural language processing (NLP) techniques to analyze construction data. For example, Shamshiri, M., Chi, S., & Moon, S. [12] reported the application of NLP to construction document analysis, while Zhang et al. [13] explored the use of text mining for automated classification of construction-related data. These studies demonstrate the potential for automation but often fail to fully incorporate hierarchical code structures or remain limited to specific datasets.

Against this backdrop, this study positions the BOQ–WCC matching task as a hierarchical text classification problem and applies an ensemble-based approach to systematically evaluate performance improvements.

3 STANDARD CONSTRUCTION WORK CLASSIFICATION CODE STRUCTURE

In Korea, construction work codes were developed to standardize BOQ by benchmarking UniClass from Europe and MasterFormat from North America. The standard WCC is a construction information classification system established by the Korea Public Procurement Service to systematically organize construction items in BOQ and standardize cost estimation. The WCC consists of a 12-digit alphanumeric code, where each digit carries a specific meaning, reflecting the hierarchical structure of the classification [6].

The first digit of the WCC represents the type of construction work. The second to sixth digits classify the work into major, medium, minor, detailed, and sub-detailed categories, distinguishing major work types, item names, and specifications. The seventh and subsequent digits provide additional information to specify particular materials or configurations. This hierarchical structure allows the WCC to further refine construction work classification from broader categories at the top level to more detailed categories at the lower levels.

As shown in the Tab. 1, the number of classes increases exponentially as the level deepens. In particular, Levels 4 and 5 primarily capture specification-level details, where data becomes sparse and highly imbalanced.

Table 1 Summarizes the number of classes at each level in the full WCC system

Level	Definition	Number of Classes
Level 1	Major work type	20
Level 2	Intermediate work type	131
Level 3	Minor work type	389
Level 4	Detailed work type	939
Level 5	Sub-detailed classification (e.g., specifications)	1,603

As shown in Fig. 1, for the purposes of this study, only the second to sixth digits of the WCC were used, corresponding to Levels 1 through 5. The first digit, which simply distinguishes between broad categories of construction work, was excluded. This approach ensures that the hierarchical structure of the WCC is maintained while optimizing analytical efficiency, as most construction items can be effectively categorized within this five-digit framework. Importantly, item identification in actual BOQs is most commonly performed up to Level 3, whereas Levels 4 and 5 represent detailed specification-level distinctions. Accordingly, performance analysis in this study was conducted with these structural characteristics in mind.

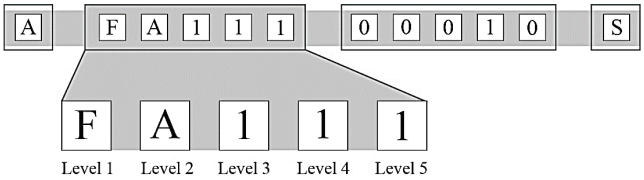

Figure 1 Structure of the Standard Construction Work Classification Code and Selected 5-Digit Levels for Analysis

4 METHODOLOGY

4.1 Dataset and Preprocessing

For this study, six BOQs from school construction projects commissioned by the Korea Public Procurement Service (PPS) were collected to construct the dataset for automatic matching between the Standard Work Classification Code (WCC) and BOQ items. The original BOQs contained approximately 12,000 entries, and after removing duplicates, 8,021 valid items were retained. Each item consists of textual information such as the major category, item description, and specifications, along with the corresponding WCC label. As shown in Tab. 2, for instance, an item categorized under Plastering Works with the description Mortar Plastering is associated with the code GA112.

When classified by major category, the dataset exhibited noticeable imbalance. Categories such as Windows and Glass Works and Metal Works accounted for more than 20% of the dataset, whereas others such as Aggregate Costs, Transportation Costs, and Demolition Works represented less than 1%. Such imbalance increases the risk of overfitting toward majority classes and deteriorates classification performance for minority categories.

Table 2 Example of Hierarchical Classification of Standard WCC

Major Category	Item Description	Level 1	Level 2	Level 3	Level 4	Level 5
Plastering Works	Mortar Plastering	G	GA	GA1	GA11	GA112
Waterproofing Works	Cement Liquid Waterproofing	H	HI	HI0	HI00	HI000
Masonry Works	0.5B Bricklaying	F	FA	FA1	FA11	FA111

Table 3 Distribution of Valid Dataset by Standard Work Categories

Work Category	Count	Percentage (%)
Window and Glass Works	2,079	25.79
Metal Works	810	10.05
Common Works	569	7.06
Finishing Works	522	6.47
Plastering Works	498	6.18
Waterproofing Works	487	6.04
Temporary Works	401	4.97
Reinforced Concrete Works	376	4.66
Masonry Works	354	4.39
Ancillary Works	338	4.19
Carpentry Works	334	4.14
Tile and Stone Works	314	3.89
Painting Works	305	3.78
Roof and Gutter Works	176	2.18
Foundation and Substructure Works	148	1.84
Miscellaneous Works	136	1.69
Earth Works	114	1.41
Demolition Works	66	0.82
Transportation Cost	20	0.25
Aggregate Cost	15	0.19
Total	**8,021**	**100**

To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) was applied. Unlike simple duplication, SMOTE generates synthetic samples by interpolating between existing minority class samples, thereby balancing the data distribution more effectively. Before applying SMOTE, some categories accounted for less than 5% of the dataset; after application, the minimum share increased to approximately 8–10%. This adjustment mitigated class imbalance and reduced training bias.

4.2 Research Workflow and Hierarchical Ensemble Model Design

The proposed methodology followed four sequential stages: data collection and preprocessing, feature extraction, hierarchical ensemble model training, and performance evaluation. The entire workflow, from dataset construction to results analysis, is illustrated in Fig. 2.

The Standard WCC is hierarchically structured, with categories becoming increasingly granular from higher to lower levels. Thus, rather than employing a single flat classifier, a hierarchical approach was adopted to reflect varying levels of complexity. Accordingly, the classification was performed progressively from Level 1 to Level 5.

Text preprocessing is a critical step in determining model quality [14]. In this study, the input features were constructed by combining major category and item description. A synonym dictionary was applied to normalize equivalent terms, and unnecessary symbols and whitespace were removed. The Keras Tokenizer was used to limit the vocabulary size to 5,000 tokens, after which texts were converted into integer sequences and padded to a maximum length of 26 tokens. For text vectorization, a Word2Vec-based embedding layer was applied, generating 128-dimensional embedding vectors. The embedding weights were updated during training, allowing the model to learn distributed representations optimized for BOQ text. Compared to frequency-based representations, this embedding approach more effectively captured semantic similarities between terms. The resulting embedding vectors were then fed into an LSTM network to capture sequential context, after which ensemble methods were applied according to the characteristics of each classification level.

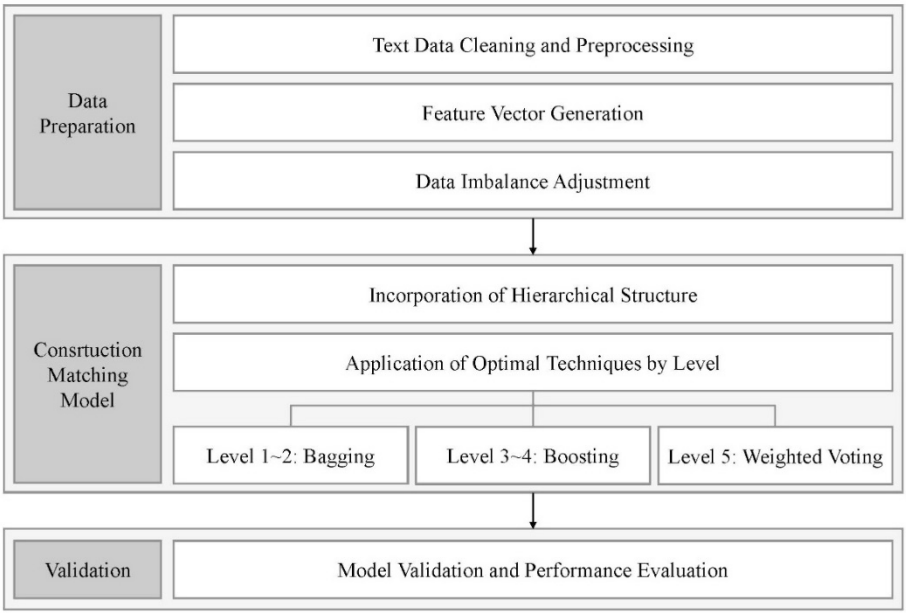

Figure 2 Framework of the Hierarchical Construction Code Matching Model

4.3 Bagging-Based Model

Bagging (Bootstrap Aggregating) is an ensemble learning technique that trains multiple predictive models and derives the final prediction through majority voting (for classification tasks) or averaging (for regression tasks). According to Breiman et al. [15], bagging reduces variance and enhances model stability by ensuring that individual models are trained independently.

In this study, bagging was applied to Level 1 and Level 2. At these upper levels, the classification scope is broader, and the dataset contains a larger volume of samples, allowing individual models to learn diverse patterns and achieve stable performance. To implement this, a Random Forest-based bagging model was employed. Random Forest operates by training multiple decision trees and aggregating their predictions to generate the final output [16].

The Random Forest-based bagging approach ensures robust predictions by mitigating data variability and outperforming single models in terms of generalization. Additionally, since each model within the ensemble is trained independently, bagging effectively prevents overfitting, further contributing to classification accuracy at Level 1 and Level 2.

4.4 Boosting-Based Model

Boosting is an ensemble learning technique that sequentially trains multiple weak learners to progressively enhance predictive performance. According to Yaman et al. [17], boosting corrects the errors of previous models, making it an effective approach for constructing a highly accurate predictive model.

In this study, boosting was applied to Level 3 and Level 4. At these lower levels, construction work classification becomes more granular, increasing the complexity of classification. Therefore, boosting was deemed appropriate as it incrementally improves the learning performance of individual models. To implement this, the XGBoost (eXtreme Gradient Boosting) algorithm was employed to develop the boosting-based model. XGBoost follows the gradient boosting framework, where misclassified samples from previous iterations are assigned higher weights so that subsequent models focus on correcting these errors [18]. In this study, the boosting-based model achieved a higher F1-score compared to individual models, demonstrating improved classification performance.

Since boosting adapts to data characteristics during training, it is particularly effective for handling higher classification complexity at Level 3 and Level 4. However, boosting models are prone to overfitting, so regularization parameters were adjusted to optimize model complexity and prevent excessive variance.

4.5 Weighted Voting-Based Model

Voting is an ensemble learning technique that combines multiple models to generate the final prediction. In majority voting (MV), all classifiers are assigned equal weights, whereas weighted voting (WV) adjusts model-specific weights based on their reliability [19]. According to Livieris et al. [20], weighted voting is an effective technique that leverages the predictive power of individual classifiers to enhance classification accuracy.

In this study, weighted voting was applied at Level 5, the most granular classification stage in the WCC hierarchy. At this level, the reliability of individual model predictions tends to decline due to the complexity and sparsity of the data. Therefore, an ensemble of multiple models was trained, and the final prediction was made using weighted voting, with the assigned weights reflecting each model's reliability. To assign weights, an F1-score-based weighting strategy was adopted. Specifically, models with higher F1-scores were assigned greater weights according to performance evaluation metrics. In this study, bagging, boosting, and logistic regression models were combined, and their weights were adjusted proportionally to their F1-scores to optimize prediction performance. Experimental results showed that the weighted voting approach achieved a higher F1-score than individual models, effectively compensating for performance discrepancies among different classifiers. Weighted voting is particularly effective in scenarios where data sparsity is high and performance variance among individual models is significant. By implementing weighted voting at Level 5, this study aimed to enhance model reliability and achieve more accurate construction code matching results.

5 RESULTS AND DISCUSSION

In this study, a hierarchical ensemble model was applied to perform automatic matching between the standard WCC and BOQ items. The model's performance was evaluated at each level using F1-score, Precision, and Recall. Additionally, the early stopping points were analyzed to determine the optimal epoch values at which the model weights were saved.

5.1 Performance Evaluation of Level 1 and Level 2

At Levels 1 and 2, a bagging-based ensemble method was applied. Tab. 4 presents the performance evaluation results, where F1-score, Precision, and Recall were used as the primary assessment metrics. At Level 1, the model achieved an F1-score of 0.9805, Precision of 0.9869, and Recall of 0.9762, demonstrating highly accurate classification performance. Early stopping occurred at epoch 16, with the optimal model weights saved at epoch 13, indicating that the model completed training at an appropriate point. The high performance at Level 1 can be attributed to clear distinctions between classes and the absence of severe class imbalance issues. In addition, the relatively balanced class distribution at this level allowed the bagging method to stabilize learning by reducing variance across individual learners, which contributed to consistent improvements in precision and recall.

Similarly, at Level 2, the bagging-based ensemble learning approach was employed, achieving an F1-score of 0.8728, Precision of 0.8799, and Recall of 0.8747. Early

stopping was triggered at epochs 8 and 11, with the optimal weights stored at epochs 5 and 8. While the performance was slightly lower than at Level 1, this decline is likely due to the increased number of classes that needed to be classified. Despite the decrease, the application of bagging reduced the risk of overfitting, which is often observed in single models at this stage, thereby maintaining stable performance even with moderately increased complexity.

Table 4 Level 1 and Level 2 Performance Metrics

Level	F1-score	Precision	Recall
1	0.981	0.987	0.976
2	0.873	0.880	0.875

5.2 Performance Evaluation of Level 3 and Level 4

At Levels 3 and 4, a boosting-based ensemble method was applied to evaluate classification performance. As shown in Tab. 5, Level 3 achieved an F1-score of 0.7423, Precision of 0.7496, and Recall of 0.7549. At this level, the number of classes increased to 548, significantly increasing classification complexity compared to Levels 1 and 2. However, the model maintained relatively stable performance, which can be attributed to the error correction mechanism of the boosting technique. Since boosting improves predictions by iteratively correcting errors from previous models, it effectively retained high classification accuracy even under complex classification conditions. Early stopping occurred at epoch 14, with the optimal weights saved at epoch 11, indicating that the model converged at an appropriate point. Nevertheless, the class distribution at Level 3 showed moderate imbalance, and boosting's ability to focus on misclassified samples proved advantageous in mitigating the underperformance of minority classes.

Similarly, at Level 4, the boosting-based ensemble method was employed, achieving an F1-score of 0.4655, Precision of 0.4645, and Recall of 0.4991. At this level, the number of classes increased to 1,490, approximately three times that of Level 3, making classification significantly more challenging. Performance declined considerably due to the increased complexity, and the model struggled to effectively learn certain classes. Early stopping occurred at epoch 9, with the optimal weights saved at epoch 6, suggesting that while the model reached optimal performance in a relatively short training period, performance degradation was observed as classification difficulty increased. The sharp drop in performance can be attributed not only to the explosion in the number of classes but also to severe class imbalance, where many categories were underrepresented. This indicates that boosting alone was insufficient to fully overcome the sparsity problem at deeper levels.

Table 5 Level 3 and Level 4 Performance Metrics

Level	F1-score	Precision	Recall
3	0.742	0.750	0.755
4	0.466	0.465	0.499

5.3 Performance Evaluation of Level 5

At Level 5, the weighted voting-based ensemble method was applied. As shown in Tab. 6, the model achieved an F1-score of 0.0001, Precision of 0.0001, and Recall of 0.0011. Early stopping occurred at epoch 4, with the optimal weights stored at epoch 1. The model encountered increasing classification complexity due to the large number of classes and insufficient samples per class, making it difficult to distinguish between classes effectively. Consequently, learning was not successfully completed, and the model struggled to capture meaningful patterns.

Furthermore, despite applying the weighted voting approach, the low performance of individual models limited its effectiveness. Weighted voting aggregates predictions by assigning higher weights to more reliable models, but at Level 5, individual models failed to achieve meaningful accuracy, making it difficult to improve final predictions through weight adjustments.

During training, validation accuracy remained close to zero, indicating a severe imbalance between training and validation data. Additionally, early stopping occurred at an initial stage, preventing sufficient learning, and the model exhibited rapid convergence without significant loss reduction during training.

This near-zero performance at Level 5 should be regarded as a critical limitation of the proposed model, reflecting the extreme sparsity and over-fragmentation of classes at this level. However, since most BOQ classifications in practice are determined within Levels 1–3, the limited applicability of Level 5 does not invalidate the practical contribution of this study. Instead, it highlights the need for future research to explore larger datasets, external specification libraries, or transfer learning techniques to address this bottleneck.

Table 6 Level 5 Performance Metrics

Level	F1-score	Precision	Recall
5	0.0001	0.0001	0.0011

5.4 Summary Analysis

The overall comparison of F1-score, Precision, and Recall across all levels is shown in Fig. 3. Overall, all performance metrics exhibit a gradual decline as the level increases. At Level 1, the model demonstrates excellent performance with an F1-score of 0.981, Precision of 0.987, and Recall of 0.976, indicating stable and reliable classification. At Level 2, the performance remains strong with an F1-score of 0.873, Precision of 0.880, and Recall of 0.875. However, a noticeable decline occurs at Level 3, where the F1-score drops to 0.742, Precision to 0.750, and Recall to 0.755. A further significant degradation is observed at Level 4, with an F1-score of 0.466, and at Level 5 the performance nearly collapses, showing that the model struggles to effectively distinguish fine-grained categories.

In addition, to validate the effectiveness of the proposed ensemble approach, a baseline comparison with a single LSTM model was conducted using the same dataset of 8,021

items. The overall comparison of F1-score changes of ensemble models relative to the base LSTM model across all levels is presented in Tab. 7. The results confirmed that ensemble methods consistently outperformed the single model across most levels, although the degree of improvement varied depending on classification complexity. Bagging provided moderate but stable improvements, with gains of approximately 3–5% at Levels 1–2 and up to 10% at Levels 3–4, but only marginal effects at Level 5. Boosting achieved the most substantial improvements, with F1-score increases of about 15% at Level 1, 18% at Level 2, and up to 20% at Level 3, although its effectiveness diminished at Level 5 due to severe class sparsity. Weighted Voting, by contrast, did not yield consistent gains and in some cases underperformed relative to the baseline, particularly at Level 5 where extreme data sparsity remained a critical challenge.

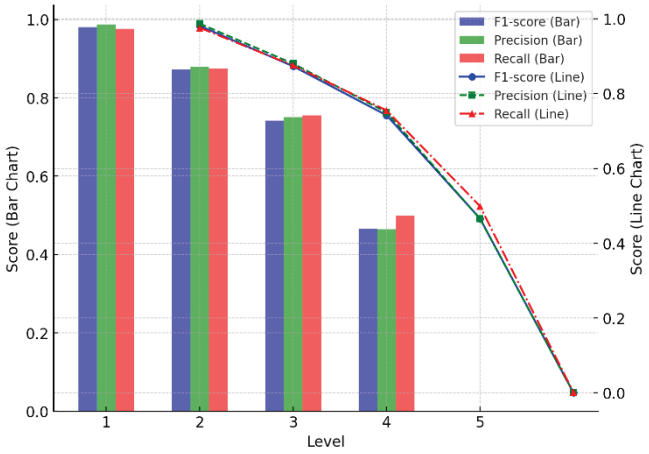

Figure 3 Comparison of F1-score, Precision, and Recall by Level

Table 7 F1-score changes of ensemble models relative to the base LSTM model

Ensemble method	Level 1 (%)	Level 2 (%)	Level 3 (%)	Level 4 (%)	Level 5 (%)
Bagging	3.5	4.2	2.0	1.5	0.0
Boosting	5.0	6.0	15.0	12.0	0.0
Weighted Voting	0.5	0.3	–0.5	–0.8	0.0

These findings demonstrate that while hierarchical ensemble learning is effective in handling multi-level construction code classification, its benefits are uneven across levels. The upper and middle levels (Levels 1–4) showed notable improvements compared to the single model, whereas Level 5 performance remained limited, highlighting the need for additional methodological enhancements and more robust datasets to overcome data sparsity in fine-grained classification.

From a practical perspective, these findings suggest that the hierarchical ensemble model can serve as a useful decision-support tool in construction project management. In particular, the stable performance observed at Levels 1–3 indicates that the model can reliably support the classification of major and medium-level work categories, which are the levels most frequently referenced in cost estimation and project planning. However, the near-zero performance at Level 5 highlights a significant limitation: when applied directly to detailed specifications or highly granular categories, the current approach may fail to provide meaningful guidance. This implies that while the proposed method has potential for practical adoption in tasks such as BOQ standardization and cost control, additional methodological enhancements and larger, more balanced datasets are required to ensure robustness at the most detailed levels.

6 CONCLUSION

To address this issue, this study aimed to develop an automated matching model between BOQ items and the standard WCC using advanced machine learning techniques. Additionally, a hierarchical ensemble learning approach was proposed to incorporate the structured characteristics of the standard WCC into the model.

In this study, a hierarchical classification model was constructed using Bagging, Boosting, and Weighted Voting techniques. Bagging-based Random Forest models were applied at Levels 1 and 2, ensuring stable classification performance. Boosting-based XGBoost models were employed at Levels 3 and 4 to improve classification accuracy in more complex classification tasks. At Level 5, Weighted Voting was implemented to aggregate model predictions based on individual model reliability.

Each ensemble method demonstrated varying performance depending on data complexity and classification level. The experimental results showed high classification accuracy at Level 1 (F1-score: 0.9805) and Level 2 (F1-score: 0.8728). Level 3 maintained relatively stable performance (F1-score: 0.7423), but performance degradation was observed at Level 4 (F1-score: 0.4655) due to increased classification complexity. Level 5 exhibited a significant decline in accuracy (F1-score: 0.0001), indicating that classification performance deteriorated as the number of classes increased and data sparsity intensified.

These findings highlight the importance of selecting appropriate ensemble techniques at different levels to optimize model performance while also emphasizing the need for improvement at lower levels. The hierarchical ensemble learning approach used in this study demonstrated high practical applicability at higher levels, particularly with Bagging, which provided stable predictions. However, as the number of classes increased at Level 3 and beyond, performance declined due to overfitting and data imbalance. Level 5 faced severe data sparsity issues, limiting the effectiveness of Weighted Voting in improving classification accuracy.

From a practical perspective, however, Levels 1–3 cover the majority of BOQ classification tasks in real projects, meaning that the proposed model can already be used to automatically assign codes to uncoded BOQ items and assist in cost verification during the design stage. Nonetheless, the near-zero performance at Level 5 represents a critical research limitation. Although its practical impact is relatively minor, as fine-grained specifications are less frequently required in BOQ preparation, overcoming this limitation remains essential for the long-term advancement of fully automated classification.

Future work will therefore prioritize: (1) expanding the dataset with larger and more diverse BOQ samples, (2) applying pre-trained embeddings such as BERT or extended Word2Vec for richer semantic representation, (3) refining the WCC hierarchy and integrating external datasets to enhance coverage, and (4) validating the model's deployment within real systems used by public agencies and contractors. By addressing these aspects, the proposed model can evolve into a fully operational tool for automated BOQ standardization and cost management, ultimately enhancing transparency and efficiency in construction project delivery.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00338908).

7 REFERENCES

[1] Noh, H. R., & Yun, S. H. (2017). A case study on the educational facility project for the improvement of work item's structure in BoQ. *Journal of the Architectural Institute of Korea Structure & Construction*, *33*(8), 47–54. https://doi.org/10.5659/JAIK_SC.2017.33.8.47

[2] Bikçe, M., & Göçer, S. (2018). An approach to preparing a bill of quantities. *International Advanced Research Engineering Journal*, *2*(3), 229–233. Available from https://dergipark.org.tr/en/pub/iarej/issue/40961/407726

[3] Cho, H. H., Kang, T. K., Lee, Y. S., & Cho, M. Y. (1999). A study on the reformation of the BoQ structure for public building projects in Korea. *Journal of the Architectural Institute of Korea Structure & Construction*, *15*(9), 123–131. Available from https://www.riss.kr/link?id=A3397343

[4] Lee, G. E., Lee, G. T., Chi, S. H., & Oh, S. O. (2023). Automatic classification of construction work codes in bill of quantities of national roadway based on text analysis. *Journal of Construction Engineering and Management*, *149*(2), 04022163. https://doi.org/10.1061/JCEMD4.COENG-12730

[5] Park, H. P., & Lee, J. S. (2004). A promotion plan through measuring the utilization of information classification systems in the construction industry. *Korean Journal of Construction Engineering and Management*, *5*(6), 90–100. UCI: G704-001084.2004.5.6.021

[6] Yun, S. H. (2010). A study on the standardized cost code system for BoQ for efficiency of cost management in public construction projects. *Journal of the Architectural Institute of Korea Structure & Construction*, *26*(12), 167–174. Available from https://www.kci.go.kr/kciportal/ci/sereArticleSearch/ciSereArtiView.kci?sereArticleSearchBean.artiId=ART001506512

[7] Guven, G., Arceo, A., Bennett, A., Tham, M., Olanrewaju, B., McGrail, M., Isin, K., & Olson, A. (2022). A construction classification system database for understanding resource use in building construction. *Scientific Data*, *9*, 99. https://doi.org/10.1038/s41597-022-01141-8

[8] Pupeikis, D., Navickas, A. A., Klumbyte, E., & Seduikyte, L. (2022). Comparative study of construction information classification systems: CCI versus Uniclass 2015. *Buildings*, *12*(5), 656. https://doi.org/10.3390/buildings12050656

[9] Danusevics, M., Braslina, L., Skiltere, D., Batraga, A., Salkovska, J., Legzdina, A., & Kalkis, H. (2022). Comparative analyses of construction classification systems in a context of benefits, challenges and required resources. In W. Karwowski, H. Kalkis, & Z. Roja (Eds.), *Social and occupational ergonomics. AHFE International Conference 2022* (p. 65). AHFE Open Access. https://doi.org/10.54941/ahfe1002654

[10] Pérez-García, A., Martín-Dorta, N., & Aranda, J. Á. (2021). BIM requirements in the Spanish public tender—analysis of adoption in construction contracts. *Buildings*, *11*(12), 594. https://doi.org/10.3390/buildings11120594

[11] Ekholm, A., & Fridqvist, S. (1996). A conceptual framework for classification of construction works. *ITcon*, *1*, 25–50. Available from https://www.itcon.org/1996/2

[12] Shamshiri, A., Ryu, K. R., & Park, J. Y. (2024). Text mining and natural language processing in construction. *Automation in Construction*, *158*, 105200. https://doi.org/10.1016/j.autcon.2023.105200

[13] Zhang, F., Fleyeh, H., Wang, X., & Lu, M. (2019). Construction site accident analysis using text mining and natural language processing techniques. *Automation in Construction*, *99*, 238–248. https://doi.org/10.1016/j.autcon.2018.12.016

[14] Hwang, J. M., Hong, S. B., & Kang, C. S. (2024). NAVER news data text mining analysis: Focusing on the keyword 'algorithm'. *Journal of Innovation in Industrial Technology*, *2*(1), 1–7. https://doi.org/10.60032/JIIT.2024.2.1.1

[15] Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

[16] Btoush, E., Zhou, X., Gururajan, R., Chan, K. C., & Alsodi, O. (2025). Achieving excellence in cyber fraud detection: A hybrid ML+DL ensemble approach for credit cards. *Applied Sciences*, *15*(3), 1081. https://doi.org/10.3390/app15031081

[17] Yaman, E., & Subasi, A. (2019). Comparison of bagging and boosting ensemble machine learning methods for automated EMG signal classification. *BioMed Research International*, *2019*, 9152506. https://doi.org/10.1155/2019/9152506

[18] Dai, Y., Xie, Y., Zhang, C., & Liu, J. (2025). Time-generative adversarial networks enabled ensemble prediction method for energy consumption of machine tools. *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2025.3534432

[19] Gokalp, O., & Tasci, E. (2019). Weighted voting based ensemble classification with hyper-parameter optimization. In *IEEE Innovations in Intelligent Systems and Applications Conference (ASYU2019)*, 1–4. https://doi.org/10.1109/ASYU48272.2019.8946373

[20] Livieris, I. E., Kanavos, A., Tampakas, V., & Pintelas, P. (2019). A weighted voting ensemble self-labeled algorithm for the detection of lung abnormalities from X-rays. *Algorithms*, *12*(3), 64. https://doi.org/10.3390/a12030064

Authors' contacts:

Yeong-Chae Yun, Master's Student
Department of Architecture Engineering,
Gyeongsang National University,
501 Jinjudaero, Jinju-si, Gyeongsangnam-do, 52828, South Korea
qwasok1029@gnu.ac.kr

Seok-Heon Yun, Professor
(Corresponding author)
Department of Architecture Engineering,
Gyeongsang National University,
501 Jinjudaero, Jinjusi, Gyeongsangnam-do, 52828, South Korea
Mobile Phone: +82-010-8860-5626
gfyun@gnu.ac.kr