Field-aware Factorization Machines for Item Recommendation in Multiplayer Online Battle Arena Games

Hyeong-Gyu Jang
School of Artificial Intelligence
University of Science and Technology
Daejeon, South Korea
brothergyu98@etri.re.kr

Sang-Kwang Lee
Electronics and Telecommunications Research Institute
Daejeon, South Korea
sklee@etri.re.kr

***Abstract*—In Multiplayer Online Battle Arena (MOBA) games, item recommendation plays a crucial role in supporting players' strategic decision making. Unlike conventional recommendation tasks, item selection in MOBA is highly context-dependent, influenced by the champion identity, positional role, current inventory, and the compositions of both allies and enemies. In this work, we propose a context-aware recommendation framework based on Field-aware Factorization Machines (FFM), which effectively captures feature interactions across heterogeneous fields. To generate training data suitable for ranking, we design a negative sampling strategy that contrasts the ground-truth item against multiple randomly sampled alternatives. Experiments demonstrate that our approach outperforms conventional baselines in ranking metrics such as Hit@K, NDCG@K, and MRR, thereby validating the utility of FFM in modeling complex contextual dependencies in MOBA environments.**

***Index Terms*—recommender systems, factorization machines, item recommendation, multiplayer online battle arena, moba**

I. Introduction

Item recommendation in Multiplayer Online Battle Arena (MOBA) games has emerged as a vital research area in both the game industry and academic communities. Unlike traditional recommendation systems where user–item interactions are relatively static, MOBA item choices are made under dynamic and multifaceted contexts. A player's decision to purchase a particular item depends on multiple factors, including the champion identity, positional role, evolving team compositions, and previously acquired items. These intricate dependencies pose significant challenges for building effective recommendation systems.

To address these challenges, we explore Field-aware Factorization Machines (FFM) as the foundational model for our recommendation framework. A key advantage of FFM is its ability to model field-specific feature interactions, making it particularly suitable for heterogeneous game contexts. For instance, our model distinguishes between fields such as the *champion field*, *ally team field*, *enemy team field*, and *candidate item field*, allowing it to learn nuanced interactions that conventional factorization or deep learning models may overlook.

This work provides three main contributions. First, we propose a structured field representation of complex MOBA gameplay situations, capturing champion identity, team compositions, and item context. Second, we introduce a ranking-oriented negative sampling strategy tailored to the recommendation task. Finally, through comprehensive experiments, we demonstrate that our FFM-based approach effectively models contextual signals and achieves competitive performance across standard ranking metrics.

Our findings demonstrate that field-aware models can simultaneously achieve accuracy, interpretability, and scalability in context-dependent recommendation tasks. In particular, by validating their effectiveness in the complex domain of MOBA item recommendation, this study highlights the significant potential of these methods for advancing intelligent systems in game analytics.

II. Related Works

A. Factorization Machines

Rendle [1] first proposed Factorization Machines (FM) as a general predictor for modeling pairwise feature interactions in sparse high-dimensional data. By combining the strengths of polynomial regression with the parameter efficiency of matrix factorization, FM provides a flexible and scalable framework that is particularly effective in recommendation and ranking tasks. Each feature is represented by a latent vector, and their pairwise inner products approximate interaction effects, thereby alleviating sparsity issues. FM has demonstrated strong performance in diverse domains such as click-through rate prediction, rating prediction, and personalized recommendation.

B. Field-aware Factorization Machines

Juan et al. [2] extended FM by proposing FFM, which introduces field-specific embeddings to enhance modeling capacity. In FFM, each feature is associated with different latent vectors depending on the target field with which it interacts. This design allows the model to capture heterogeneous relationships more precisely, such as distinguishing interactions

between user–item pairs from those between item–context or item–position pairs. FFM has achieved state-of-the-art results in large-scale recommendation benchmarks, including CTR prediction competitions, and has been widely adopted in industry applications for its superior capacity to handle complex categorical structures.

C. Recommendation in MOBA Games

Recent studies have examined recommendation in MOBA games [3], [4], a domain where item choices are highly dynamic and context-dependent. Unlike conventional recommendation scenarios, a player's decision is influenced not only by their own champion and positional role but also by the evolving composition of allies and enemies, as well as previously acquired items. Various machine learning approaches, ranging from collaborative filtering to deep learning, have been applied to model these complex dependencies. However, field-aware models such as FFM remain underexplored in this domain, despite their strong potential for capturing heterogeneous contextual signals. Our work bridges this gap by applying FFM to MOBA item recommendation, demonstrating its effectiveness in modeling the interplay between champion identity, team composition, and item context.

III. Proposed Method

A. League of Legends

League of Legends is a team-based strategy game in which two teams (Blue and Red), each consisting of five champions, compete to destroy the opponent's base, known as the Nexus.

A unique characteristic of this game is that champion performance is strongly influenced by item acquisition during a match. Items provide not only raw statistical enhancements (e.g., attack power, defense, mobility) but also situational utilities such as healing reduction, vision control, or crowd-control effects. Consequently, item purchase decisions directly affect the outcome of lane skirmishes, team fights, and ultimately, the game itself. Unlike static role assignments or fixed champion abilities, the evolving sequence of purchased items introduces a highly dynamic decision-making process. This characteristic makes League of Legends a natural yet challenging domain for item recommendation research. An effective system must therefore account for numerous factors, including champion identity, positional role, team compositions, and the temporal sequence of prior purchases.

B. Dataset

Our dataset was constructed from match data from League of Legends patch 15.15. A total of 12,074 high-ranked solo queue games (Challenger, Grandmaster, and Master tiers) were collected in JSON format via the official Riot Games API. From these match logs, we extracted minute-by-minute item purchase and sale events, which were then aggregated into a single DataFrame.

To contextualize each recommendation instance, we enriched the event logs with team compositions and player roles. Each record was subsequently structured into the following fields:

- **ChampionId (Champ):** The champion played by the focal player.
- **ContextKey (Ctx):** The sequence of items the player had acquired up to that point.
- **CandidateItem (Cand):** The next item purchased, serving as the prediction target.
- **AllyComp (Ally):** The set of allied champions in the match.
- **EnemyComp (Enemy):** The set of enemy champions in the match.
- **PositionId (Pos):** The positional role of the player (e.g., top, jungle, mid, bottom, support).

To ensure consistent model training, we standardized all champion and item identifiers. Specifically, the original in-game keys (e.g., a champion with ID 711) were remapped to a compact categorical index (e.g., $711 \rightarrow 155$). This preprocessing step ensured that all 171 unique champions and 214 unique items were represented by consecutive integer IDs, which facilitates efficient embedding and field-aware modeling.

Through this pipeline, we obtained a large-scale, training-ready dataset of approximately 1.65 million interaction instances. This dataset robustly captures the heterogeneous contextual factors influencing item decisions, providing a solid foundation for our field-aware recommendation models.

C. Field-aware Factorization Machines Modeling

The proposed recommendation framework is based on the FFM, which is particularly suitable for capturing heterogeneous contextual signals in MOBA environments. Unlike conventional factorization models that assign a single embedding to each feature, FFM allocates distinct embeddings depending on the interacting field. This allows the model to capture subtle relationships such as champion–ally, champion–opponent, and item–context interactions that arise in dynamically evolving game states.

In our formulation, each categorical input field (champion, position, context, ally, enemy, and candidate item) is encoded into a field-specific embedding tensor. These embeddings are then processed through two complementary components: (i) a pairwise interaction layer that models second-order feature interactions, and (ii) a linear layer that aggregates global biases and individual feature contributions. The outputs of these two components are combined and passed through a sigmoid activation function to estimate the probability that a candidate item will be selected in the next step.

This design provides three key advantages for MOBA item recommendation. First, field-specific embeddings enable the model to clearly distinguish between different types of interactions, such as the synergy effects that certain champions exhibit within allied team compositions. Second, the factorization structure ensures parameter efficiency, allowing the model to scale to hundreds of champions and items without overfitting. Third, the parallel linear component preserves the direct contribution of individual features, thereby enhancing

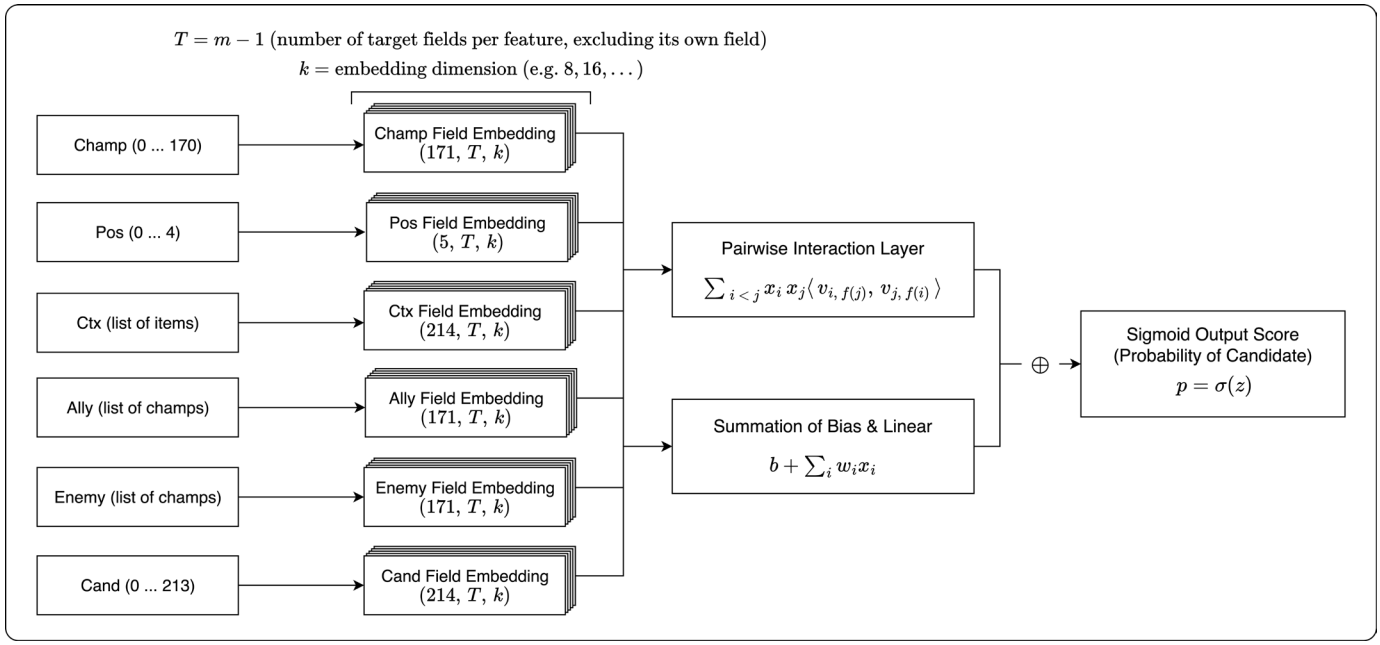

Fig. 1. Schematic illustration of the proposed field-aware factorization machine architecture. Each categorical input field (Champion, Position, Context, Ally, Enemy, and Candidate) is mapped to a field-specific embedding tensor of shape (size, T, k), where $T = m-1$ denotes the number of target fields per feature (excluding its own field) and k is the embedding dimension. The embeddings are used to compute pairwise interaction terms, $\sum_{i<j} x_i x_j \langle v_{i,f(j)}, v_{j,f(i)} \rangle$, while a parallel linear component aggregates the global bias and weighted features, $b + \sum_i w_i x_i$. The two contributions are merged ($\oplus$) into the final logit z, which is passed through a sigmoid function to yield the candidate probability, $p = \sigma(z)$.

interpretability. Taken together, these characteristics establish FFM as a scalable and interpretable approach for modeling complex dependencies in MOBA item recommendation.

IV. Experimental Results

We evaluate the proposed FFM on the processed League of Legends dataset. The experiments focus on the effect of embedding dimensionality $k \in \{8, 16, 32, 64\}$, and on the negative sampling strategy employed during training and evaluation.

A. Training and Evaluation Protocol

We adopt a ranking-based negative sampling approach for both training and evaluation. During training, for each ground-truth instance (the true next item), we sample $K_{\text{neg\_train}} = 50$ negative items to form the candidate set. This approach balances computational efficiency with the need for the model to learn to discriminate between relevant and irrelevant items. For evaluation, however, we employ a more rigorous protocol to simulate a realistic scenario. The candidate set for each prediction is expanded to include all 213 other items available in the game. This requires the model to rank the true item against every possible alternative, providing a comprehensive test of its ranking capability.

B. Evaluation Metrics

We employ multiple ranking-based metrics, each capturing complementary aspects of recommendation quality:

- **Accuracy@1 (Acc)**: proportion of cases where the top-ranked item is the ground-truth choice. This is the strictest evaluation of prediction correctness.
- **Mean Reciprocal Rank (MRR)**: averages the reciprocal rank of the true item. It rewards predictions that rank the correct item near the top.
- **Hit@K**: checks whether the ground-truth item appears within the top-K predictions. This reflects practical usability when players are recommended a short list.
- **Normalized Discounted Cumulative Gain(NDCG)@K**: measures ranking quality with position-based discounts, giving higher credit to correctly ranked items near the top of the recommendation list [5].

The combination of these metrics ensures a comprehensive evaluation: strict correctness (Accuracy), overall ranking quality (MRR, NDCG), and practical recommendation utility (Hit).

C. Performance with Different Embedding Dimensions

Table I summarizes the performance of FFM with different embedding dimensions (k). The results indicate that increasing the embedding dimension from $k = 8$ to $k = 16$ yields a notable performance improvement across all metrics. For instance, Accuracy@1 rises from 0.3204 to 0.3504, and NDCG@10 improves from 0.5932 to 0.6243. Further enlarging the embedding dimension to $k = 32$ and $k = 64$ provides only marginal gains, with Accuracy@1 converging around 0.3564 and NDCG@10 stabilizing near 0.6335. These results suggest that while higher embedding dimensions can capture richer interactions, the performance gains saturate beyond $k = 16$.

TABLE I
FFM PERFORMANCE WITH DIFFERENT EMBEDDING DIMENSIONS (k).

Dim k	Acc@1	MRR	Hit@3	Hit@5	Hit@10	NDCG@3	NDCG@5	NDCG@10	#Params	Time/Epoch (sec)
8	0.3204	0.5058	0.6191	0.7529	0.8899	0.4933	0.5485	0.5932	**46,355**	**43.05**
16	0.3504	0.5367	0.6588	0.7925	0.9154	0.5289	0.5840	0.6243	91,763	95.02
32	0.3563	0.5447	0.6697	0.8052	0.9230	0.5379	0.5939	0.6326	182,579	128.65
64	**0.3564**	**0.5453**	**0.6713**	**0.8068**	**0.9246**	**0.5389**	**0.5948**	**0.6335**	364,211	305.10

TABLE II
EFFICIENCY BY EMBEDDING DIMENSION: METRIC-PER-SECOND AND RELATIVE EFFICIENCY (NORMALIZED TO k=8 = 100%).

Dim k	Top-1/sec	Rel. (%)	Top-K/sec	Rel. (%)
8	0.00960	**100.0**	0.01509	**100.0**
16	0.00467	48.6	0.00720	47.7
32	0.00350	36.5	0.00539	35.7
64	0.00148	15.4	0.00228	15.1

TABLE III
PARAMETER-NORMALIZED EFFICIENCY: AVERAGE PERFORMANCE PER 10^5 PARAMETERS (NORMALIZED TO k=8 = 100%).

Dim k	Top-1/Param	Rel. (%)	Top-K/Param	Rel. (%)
8	0.89117	**100.0**	1.40111	**100.0**
16	0.48336	54.2	0.74538	53.2
32	0.24674	27.7	0.37955	27.1
64	0.12379	13.9	0.19082	13.6

D. Efficiency Analysis

We evaluate efficiency based on two aggregate metrics: Top-1, defined as the average of Acc@1 and MRR, and Top-K, defined as the average of Hit@3/5/10 and NDCG@3/5/10. Table II summarizes efficiency in terms of performance per training time. $k = 8$ achieves the highest efficiency, while larger embedding dimensions incur steep drops: $k = 16$ retains only 50%, $k = 32$ 35%, and $k = 64$ merely 15% of the baseline efficiency. Although larger embeddings slightly improve accuracy, they do so at a disproportionately high computational cost. Thus, $k = 16$ provides a balanced trade-off, whereas $k = 8$ remains the most efficient choice for scalable deployment.

Table III further presents parameter-normalized efficiency. While the number of parameters increases linearly with embedding dimension, performance saturates rapidly in a logarithmic-like manner. As a result, parameter efficiency declines almost proportionally with model size: from 100% at $k = 8$ to only 54% at $k = 16$, 27% at $k = 32$, and 14% at $k = 64$. This demonstrates that larger models do not yield proportional accuracy gains, but instead suffer sharply diminishing returns.

In summary, embedding dimension scaling exhibits clear diminishing returns: accuracy improvements saturate, while training time and parameter count grow substantially. $k = 8$ and $k = 16$ emerge as the most practical trade-offs for large-scale deployment.

V. CONCLUSION AND FUTURE WORKS

In this work, we investigated the task of item recommendation in MOBA games, where decision-making is inherently context-dependent and influenced by multiple heterogeneous factors. To address the complexity of such dependencies, we proposed the use of FFM as the recommendation backbone. By leveraging field-specific embeddings, our approach effectively captures nuanced interactions across champion identity, positional role, team compositions, and item context.

In addition, this study adopted a negative sampling strategy to align with the ranking-oriented nature of the recommendation task. The proposed approach demonstrated stable and strong performance overall. Notably, while increasing embedding dimensionality is beneficial up to a certain point, further expansion yields diminishing returns in performance while sharply escalating computational costs. This highlights the necessity of carefully balancing accuracy and efficiency when deploying models in large-scale real-world environments.

Overall, our findings validate both the effectiveness and limitations of field-aware models in MOBA item recommendation. Beyond demonstrating the utility of FFM in capturing heterogeneous contextual dependencies, this study emphasizes the need to balance predictive accuracy with computational efficiency in real-world systems. Future research directions include extending the framework to sequential modeling of item purchases, integrating temporal dynamics of matches, and exploring hybrid architectures that combine field-aware factorization with deep neural models.

ACKNOWLEDGMENT

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (Project Name: Development of generative AI-based esports service automation platform technology to improve esports operation efficiency, Project Number: RS-2024-00441523, Contribution Rate: 100%)

REFERENCES

[1] S. Rendle, "Factorization Machines," *2010 IEEE International Conference on Data Mining*, Sydney, NSW, Australia, 2010, pp. 995–1000.

[2] Y. Juan, Y. Zhuang, W.-S. Chin, and C.-J. Lin, "Field-aware factorization machines for CTR prediction," *10th ACM Conference on Recommender Systems (RecSys '16)*, Boston, MA, USA, 2016, pp. 43–50.

[3] V. Araujo, F. Rios, and D. Parra, "Data mining for item recommendation in MOBA games," *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM '19)*, New York, NY, USA: ACM, 2019, pp. 2573–2581.

[4] A. Dallmann, J. Kohlmann, D. Zoller, and A. Hotho, "Sequential item recommendation in the MOBA game Dota 2," *2021 International Conference on Data Mining Workshops (ICDMW)*, 2021, pp. 10–17.

[5] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Transactions on Information Systems*, Oct. 2002, vol. 20, no. 4, pp. 422–446.