A Gated Review Attention Framework for Topics in Graph-Based Recommenders

Eduardo Ferreira da Silva*
eduardoferreira@ufba.br
Instituto de Computação, UFBA
Salvador, Bahia

Joel Pires
joelpires@ufba.br
Instituto de Computação, UFBA
Salvador, Bahia

Denis Robson Dantas Boaventura
denis.boaventura@ufba.br
Instituto de Computação, UFBA
Salvador, Bahia

Mayki dos Santos Oliveira
maykioliveira@ufba.br
Instituto de Computação, UFBA
Salvador, Bahia

Frederico Araujo Durão
fdurao@ufba.br
Instituto de Computação, UFBA
Salvador, Bahia

ABSTRACT

Recommender systems significantly reduce information overload by curating personalized content on digital platforms, thereby enhancing user experience. Traditional models often rely on sparse rating data, overlooking the rich semantic signals embedded in user reviews. To address this, we propose the Gated Review Attention Framework for Topics, a novel graph-based recommender system that integrates review-derived topic information with user-item interactions. Our approach leverages BERTopic to extract interpretable semantic topics from textual reviews and then embeds them into a graph attention network architecture. A gating mechanism dynamically regulates the influence of these topic representations relative to latent user and item embeddings, enabling adaptive feature fusion. We evaluate GRAFT on three benchmark datasets: Amazon Movies and TV, IMDb, and Rotten Tomatoes. Comparing it against classical and neural baselines, including SVD, DeepCoNN, and KANN. Experimental results demonstrate that GRAFT consistently achieves the lowest RMSE across all datasets, indicating superior rating prediction accuracy. Although traditional models perform better on ranking metrics, GRAFT achieves superior accuracy (lower RMSE), and our qualitative analysis demonstrates more substantial semantic alignment.

KEYWORDS

collaborative filtering, review-based recommendations, topic-extraction recommendation.

1 INTRODUCTION

Recommender systems have emerged as a robust solution to address the exponential growth of products, services, and online information, acting as information filters. These systems provide personalized recommendations to users based on their preferences, needs, and past behaviors [2]. Jannach et al. [12] defines Recommender System (RS) as automated software tools that leverage machine learning and statistical models to suggest items, such as products, movies, books, or music, tailored to individual user preferences. These systems have become pervasive in modern digital experiences, driving key functions across e-commerce, entertainment, and social media. By offering personalized suggestions, they enhance user satisfaction, reduce decision-making effort, and generate substantial business value through increased engagement and sales.

Additionally, Pires et al. [17], Raza and Ding [18] emphasizes the importance of RS due to their ability to deliver relevant and personalized content. While practical algorithms are essential, the core challenge involves accurately understanding and modeling user preferences [24]. RS must identify complex relationships within data, uncovering patterns and connections that are often not immediately apparent. In this context, online reviews can enhance this process by offering rich, user-generated data that captures nuanced experiences and preferences. These reviews help describe the complex interactions between users and items, providing contextual signals that go beyond structured data such as ratings or clicks. Online reviews, which users generate based on their personal experiences, have become a central channel for sharing opinions in digital environments. Unlike traditional advertising, users often perceive these reviews as more authentic and persuasive, which can influence their decisions. Favorable evaluations can boost confidence, while negative reviews usually deter potential interactions [5, 14, 21].

Beyond their credibility, reviews shape consumer behavior through emotional and social dynamics. The sentiments that consumers express can create emotional connections that reinforce user engagement. Star ratings and consensus opinions serve as social proof, guiding consumers to align with the majority view. This influence arises from a desire for accurate information and social belonging. Consumers may follow group opinions to gain approval or avoid criticism, even when the choice is not the optimal one. Moreover, reviews serve as a feedback mechanism for businesses, helping identify strengths and areas for improvement, and fostering loyalty through user engagement [5, 21].

User reviews offer nuanced opinions beyond simple numerical ratings, enabling RS to build more detailed user and item profiles [4]. Although this data is unstructured, advancements in Natural Language Processing (NLP) facilitate the extraction of key information, such as topics and sentiments, a strategy proven to improve

*Both authors contributed equally to this research.

In: Proceedings of the Brazilian Symposium on Multimedia and the Web (WebMedia'2025). Rio de Janeiro, Brazil. Porto Alegre: Brazilian Computer Society, 2025.

ISSN 2966-2753

recommendation models [8]. In this context, the integration of textual insights into RS directly addresses fundamental challenges. Rich review content helps mitigate data sparsity and cold-start problems by uncovering user preferences even when explicit ratings are missing [26]. Moreover, by anchoring recommendations in detailed user feedback, the resulting models offer greater transparency and interpretability [13]. We hypothesize that integrating semantic information from reviews into the user-item graph can help the model learn more expressive representations for both users and items.

To explore this, we propose the Gated Review Attention Framework for Topics (GRAFT), a graph-based recommender that incorporates topic extraction from user reviews to enrich these interactions. GRAFT models user-item relationships as a bipartite graph, where nodes represent users and items, and edges reflect their interactions. We integrate topic-based representations of reviews as additional signals into this structure. We apply a gated attention mechanism to balance the contribution of review-based information with user and item embeddings, allowing the model to select the most informative features adaptively. This approach enhances the representation of complex user-item relationships by combining collaborative signals with content derived from reviews.

Our work highlights several key differences from existing literature. Firstly, the reviews undergo pre-processing in a prior task. This approach enables a detailed analysis of each dataset and facilitates the search for the optimal configuration, allowing for an examination of potential human understanding. Secondly, we proposed a double attention-gated mechanism that provides two levels of decision-making regarding the selection of embeddings or the distribution of topics in the training process. The model operates at the node level, considering the internal relationships within the graph, and at the structural level, where we assign each node a greater weight to adjust the overall weights more effectively.

The main contributions of this work are as follows.

(1) We introduce a method to incorporate topic-based representations of user reviews into a recommendation model, utilizing a gated attention mechanism to balance the contributions of textual content and user-item embeddings dynamically.
(2) We present an analysis that examines the alignment between review-based topics in the ground-truth list and those in the generated recommendations, offering a qualitative perspective on the semantic consistency of the recommendations.

This paper is structured as follows. Section 2 presents the related works. Section 3 introduces the background and supporting techniques. Section 4 presents the proposed GRAFT framework, detailing its architecture and integration of review topics into the recommendation process. Section 5 describes the experimental setup, including datasets, baselines, and evaluation metrics. Section 6 provides an in-depth analysis of GRAFT's recommendation behavior, with emphasis on the composition of the ranked lists. Finally, Section 7 concludes the paper and outlines directions for future work.

2 RELATED WORKS

Several early models utilized review texts to address data sparsity and enrich user-item representations. DeepCoNN [25] introduced parallel CNN-based networks for users and items to share a layer for rating estimation. It treated reviews as rich user feedback; however, CNN's inability to model long-range dependencies limited its effectiveness. Similarly, NARRE [3] utilized CNNs for text encoding and incorporated an attention mechanism to weight informative reviews, thereby enhancing prediction and explainability. However, neither model incorporated structured semantic representations nor handled variations in review quality over time.

Other models incorporated topic or aspect-based representations to model fine-grained preferences. Musto et al. [16] extracted entities and sentiment scores from reviews to construct multi-dimensional user profiles, which we then used in a multi-criteria collaborative filtering setting. Cheng et al. [6] proposed 3NCF, which also focused on capturing aspect-level signals, employing a topic-guided attention mechanism to align user interests with item characteristics. These approaches emphasized matching user preferences with interpretable features derived from text but did not leverage graph-based relational structures.

More recent models have combined deep language representations and attention mechanisms to capture contextual information more effectively. Li et al. [14] used BiGRU and attention refine RoBERTa embeddings to emphasize salient review segments, integrating them into an Attentional Factorization Machine (AFM) to capture nuanced user-item interactions while reducing the impact of noise. A separate line of work explores graph-based methods that integrate review information with user-item interaction structures. Yang and Cai [22] presented ERRI, which models both semantic relations between review words through a review text graph and collaborative signals through a user-item interaction graph. We fuse these two sources to inform rating predictions. He et al. [10] introduced RDRec, a diffusion-based approach that simulates user interest evolution by corrupting and reconstructing review-based features using a transformer. Shang et al. [19] further explored sentiment-aware representations by applying hierarchical attention to review texts, capturing emotional nuance and integrating it into a neural collaborative filtering model.

GRAFT differs from these works by unifying topic modeling and graph attention in a gated framework. Rather than relying solely on word- or review-level attention, GRAFT encodes topical information as part of node features. It employs graph attention layers with gating mechanisms that regulate the propagation of semantic and relational signals throughout the network. This design enables more interpretable and adaptive representation learning, particularly in cases where interaction patterns are sparse or heterogeneous.

3 BACKGROUND

This work integrates three complementary techniques to enhance recommendation performance based on graph and textual data. We employ BERTopic [9] to extract high-level semantic information from user reviews by combining transformer-based embeddings with clustering techniques. This method enables the identification of coherent topics across reviews, which we can integrate into the model to enhance interpretability and user preference modeling. Second, Graph Attention Networks (GAT) [20] provide a powerful mechanism for learning on graph-structured data. Lastly, inspired by gated units in Recurrent Neural Networks [7], we incorporate a

gating mechanism that regulates the contribution of distinct information sources, such as user/item embeddings and topic features.

3.1 Topic Modeling with BERTopic

BERTopic is a neural topic modeling technique that leverages transformer-based embeddings and clustering to extract coherent topics from textual data. It begins by encoding documents using pre-trained language models, preserving semantic relationships in a high-dimensional vector space. We then project these embeddings into a lower-dimensional space and group them into semantically consistent clusters with a non-supervised algorithm. To construct interpretable topics from these clusters, BERTopic employs a class-based variant of Term Frequency-Inverse Document Frequency (TF-IDF), which identifies words most representative of each cluster. This pipeline enables the discovery of latent themes in reviews, which we integrate into the recommendation framework as interpretable and context-rich signals [9].

3.2 Graph Attention Networks

Veličković et al. [20] proposed Graph Attention Networks (GAT), which are a neural architecture that operates on graph-structured data, allowing nodes to attend to their neighbors during representation learning dynamically. Unlike traditional graph convolutional networks, which uniformly aggregate information from neighbors, GAT employs a self-attention mechanism to assign distinct importance weights to each neighbor, thereby making the aggregation process more expressive and adaptive. This is particularly suitable for recommendation tasks, as it naturally captures the collaborative filtering principle by prioritizing informative neighbor interactions during the learning process. By leveraging multi-head attention, GAT not only enhances model capacity and robustness but also retains interpretability through the learned attention weights.

3.3 Gated Units and Feature Control

Inspired by Cho et al. [7] gated mechanisms in Recurrent Neural Networks (RNNs), such as the Gated Recurrent Unit (GRU), we incorporate a gating strategy to regulate the contribution of distinct data sources in our model. In RNNs, gates control how much past information should be retained or forgotten when computing the next hidden state. Drawing from this intuition, our model adapts a similar mechanism to balance between user/item embeddings and topical features derived from textual reviews. The gate outputs a continuous value that determines the extent to which each representation, semantic or structural, contributes to the final fused representation, thus enabling a more flexible and context-aware integration during training.

4 PROPOSAL

4.1 Topic Extraction from reviews

The first step of topic extraction was the pre-processing and cleaning phase. The approach adopted preserves the full set of reviews, removing only non-informative elements such as HTML tags, excessive punctuation, and dataset-specific textual artifacts. For example, in the Amazon Movies and TV dataset, item descriptions embedded within some reviews were removed. Emojis and short reviews were retained to preserve the original characteristics of the data. In sequence, we use BERTopic to generate topics by utilizing transformer-based models to convert each document into a dense vector, thereby placing similar documents closer together in a high-dimensional space. We generate the embedding representation from the sentence transformers library `all-mpnet-base-v2`[1] Additionally, to enable clustering of high-dimensional embeddings, we apply dimensionality reduction using Uniform Manifold Approximation and Projection (UMAP). The model is configured through hyperparameters, including the number of components, distance metric, number of neighbors, and minimum distance. After reduction, the K-means clustering algorithm is used to identify clusters. This is particularly important, as not all documents are associated with a well-defined topic.

The topic representation chosen in BERTopic utilizes the main keywords associated with each topic. For topic labeling, we used the KeyBERT-Inspired representation model. This model selects representative terms for each topic by aligning them with the embedded documents in each cluster. Domain-specific stopwords (e.g., "movie," "film," "dvd," "disc") were also removed to eliminate terms with low discriminative value. Additionally, the number of features was defined, and n-gram representations (unigrams, bigrams, and trigrams) were included to enhance the expressiveness and clarity of the extracted topics. The overall BERTopic pipeline involved:

(1) Embedding reviews.
(2) Reducing embedding dimensions.
(3) Identifying topics with the clusterization method.
(4) Representing topics using KeyBERT-derived keywords.
(5) Using a custom stopword list to remove domain-irrelevant terms and trigrams for richer features.

4.2 Gated Review Attention Framework for Topics (GRAFT)

To integrate topics derived from reviews into a graph-based collaborative filtering architecture, GRAFT[2] employs a graph neural architecture that jointly leverages rating interactions and review topics through gated fusion mechanisms and attention-based message passing. This analysis revisits our initial hypothesis that incorporating semantic information from reviews into the user-item graph can enhance the expressiveness of learned representations and improve the quality of recommendations. To implement this, the model begins by assigning each user u and item i a learnable embedding vector $\mathbf{e}_u, \mathbf{e}_i \in \mathbb{R}^d$, where d is the base embedding dimension. We then project these initial representations into a structural space of size d' to align with the multi-head attention mechanism, as defined by:

$$\begin{aligned} h_u &= ReLU(W_{proj} \, .e_u), \\ h_i &= ReLU(W_{proj} \, .e_i), \end{aligned} \tag{1}$$

where the h_u and h_i are the projected embeddings of the user and items, after applying a linear transformation by a ReLu activation, and $W_{proj} \in \mathbb{R}^{d' \times d}$ is the projection matrix used to transform the initial embeddings in the structural space used in Graph Attention Network (GAT) layers.

[1]Available in: https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[2]Available at: https://github.com/ferreira-eduardo/GRAFT.git

The model interactions are the multiple layers of the GATv2Conv operator. Each layer propagates and aggregates neighbor information using edge features as attention biases,

$$h'_v = \text{GATv2Conv}(h_v, h_{\mathcal{N}(v)}, e_{uv}), \tag{2}$$

where $e_{uv} \in \mathbb{R}^k$ is the topic distribution and k the dimensionality, on the edge from node u to node v. Additionally, to define the gated mechanism, a transformation is applied to a vector of representation $\mathbf{e}_u$, which represents a user embedding within the model's context.

To control the flow of semantic information, the model employs a learned gating mechanism that regulates how much of each transformed embedding dimension is retained. The gate vector for user u is defined as:

$$\mathbf{g}_u = \sigma\big(\mathbf{W}_g^u \, \mathbf{e}_u + \mathbf{b}_g^u\big) \quad \in \ [0,1]^k, \tag{3}$$

where $\sigma(\cdot)$ is the element-wise sigmoid function, $\mathbf{W}_g^u \in \mathbb{R}^{k\times d}$ is a learned projection matrix, and $\mathbf{b}_g^u$ is the corresponding bias. The dimensionality k corresponds to the topic space.

In parallel, a linear transformation is applied to map the original user embedding into the same topic-aligned space:

$$\tilde{\mathbf{e}}_u = \mathbf{W}_p^u \, \mathbf{e}_u + \mathbf{b}_p^u \quad \in \ \mathbb{R}^k, \tag{4}$$

where $\mathbf{W}_p^u \in \mathbb{R}^{k\times d}$ and $\mathbf{b}_p^u$ are also learnable parameters. This transformed embedding $\tilde{\mathbf{e}}_u$ is then modulated by the gate $\mathbf{g}_u$, which selectively filters or amplifies individual components of the representation. This gated linear unit allows the model to dynamically control feature fusion, learning which aspects of the semantic transformation are most relevant for each user.

On the structural level, it involves blending two representations of a user or an item. We define a gated fusion mechanism that combines a topic-based vector $\mathbf{t}_x$ and a transformed embedding $\tilde{\mathbf{e}}_x$, modulated by a learned gate vector $\mathbf{g}_x$. The fusion is given by

$$\hat{\mathbf{t}}_u = \mathbf{g}_u \odot \mathbf{t}_u \ + \ \big(1 - \mathbf{g}_u\big) \odot \tilde{\mathbf{e}}_u, \tag{5}$$

where $\odot$ denotes element-wise product. This operation performs a dimension-wise blend between semantic and structural features. For each dimension j, the fusion can be expressed as

$$\hat{t}_{x,j} = g_{x,j} \, t_{x,j} + (1 - g_{x,j}) \, \tilde{e}_{x,j}, \tag{6}$$

where $g_{x,j} \in [0,1]$ controls the relative influence of the topic and structural components. When $g_{x,j} \approx 1$, the model prioritizes the topic signal, whereas when $g_{x,j} \approx 0$, it favors the projected embedding. This formulation enables adaptive feature integration, allowing the model to learn per-dimension preferences for each node type.

Finally, the model produces a prediction by combining the gated structural and topic-aware representations of the user and item. Let $\hat{\mathbf{t}}_u$ and $\hat{\mathbf{t}}_i$ denote the fused representations obtained after the final gating step. These vectors are concatenated and passed through a fully connected layer:

$$\mathbf{x} = \text{FC}\big[\, \mathbf{e}_u \,\|\, \mathbf{a}_u \,\|\, \hat{\mathbf{t}}_u \,\|\, \mathbf{e}_i \,\|\, \mathbf{a}_i \,\|\, \hat{\mathbf{t}}_i \,\big], \tag{7}$$

where $\mathbf{a}_u, \mathbf{a}_i$ are the outputs of the attention fusion submodules and $\|$ denotes concatenation. This attention-gated fusion enables the model to learn when to trust latent embeddings versus topical signals on a per-dimension basis. The concatenated vector $\mathbf{x}$ is passed through a fully connected layer with dropout, followed by a scaled activation to constrain the output to the rating range that defines the width of the rating scale. The final output is a scalar prediction of the user's rating for the item.

5 EXPERIMENTAL SETUP

5.1 Datasets

We select three distinct datasets: Amazon Movies and TV[3], IMDb[4], and Rotten Tomatoes[5]. These sources offer a comprehensive view by integrating user-generated reviews with audience ratings and metadata across diverse platforms. Table 1 presents key statistics for each dataset before and after preprocessing.

We began by removing invalid or incomplete entries from all datasets to ensure data quality. As a general criterion, we selected users with at least 10 interactions to focus on active profiles and reduce sparsity. We then further processed each dataset according to its specific characteristics. Due to its large scale, the Amazon dataset required additional filtering: we also retained only items with at least 100 interactions to ensure meaningful representation. The Rotten Tomatoes dataset posed a unique challenge, as it includes non-numeric rating formats such as "A", "B", "F", and "A-". To ensure consistency and comparability, we retained only entries with normalized fractional or numeric ratings (e.g., 1/5, 2/4, 10/10), aligning all scores to a standard 1–5 scale. Notably, the IMDb dataset experienced a substantial reduction in size after filtering, reflecting its initially sparse distribution of user-item interactions.

Table 1: Impact of preprocessing, removal of incomplete entries, retention of users with ≥ interactions (and items with ≥ 100 interactions for Amazon), and rating normalization for Rotten Tomatoes, on dataset size (reviews, users, items), reviews per user/item, and overall sparsity across the Amazon Movies & TV, IMDb, and Rotten Tomatoes datasets.

	Amazon Movies and TV		IMDb		Rotten Tomatoes	
Metric	**Raw**	**Processed**	**Raw**	**Processed**	**Raw**	**Processed**
Reviews	17.3M	1.4M	932.4K	264.9K	1.1M	543.5K
Users	6.5M	76K	427K	7k	11K	2.6k
Items	747.7K	27.3K	1.1K	1.1K	17.7K	17.4K
Reviews/User	2.664	19.12	3.28	37.07	100.06	208.17
Reviews/Item	23.173	53.18	12.28	230.40	63.77	31.13
Sparsity	99.99%	99.93%	99.81%	96.77%	99.42%	98.80%

5.2 Baselines

For comparison, we evaluate our approach against three baseline algorithms: Singular Value Decomposition (SVD)[6], DeepCoNN [25], and KANN [15]. SVD is a classical and widely adopted collaborative filtering technique that factorizes the user-item interaction matrix into lower-dimensional latent spaces. By capturing hidden patterns in user preferences and item characteristics, SVD has demonstrated strong performance in various recommendation scenarios. DeepCoNN enhances collaborative filtering by incorporating textual reviews from both users and items. It employs two parallel convolutional neural networks to learn user and item representations from

[3]Available at: https://amazon-reviews-2023.github.io/
[4]Available at: https://ieee-dataport.org/open-access/imdb-movie-reviews-dataset
[5]Available at: https://www.kaggle.com/datasets/stefanoleone992/rotten-tomatoes-movies-and-critic-reviews-dataset
[6]Available in: https://surprise.readthedocs.io/en/stable/matrix_factorization.html

review text, thereby improving rating prediction. KANN builds upon this concept by integrating review content with external knowledge through a knowledge-aware attention mechanism. This enables the model to generate more accurate recommendations while also providing interpretable explanations.

5.3 Hyperparameter Tuning

For GRAFT, we adopted an empirical approach to select hyperparameters, concentrating on those that significantly influenced model performance. Initially, we conducted randomized sampling over the following search space: embedding dimensions 32, 64, 128; number of layers 2, 3, 4; number of attention heads 2, 4, 8; leaky ReLU negative slope (α) 0.2, 0.3, 0.4, 0.5; dropout rates 0.2, 0.3, 0.4, 0.5; and learning rates 0.1, 0.001, 0.0001. From this search, we identified the best-performing hyperparameter configuration for each dataset. For the IMDB dataset, the optimal model parameters included an embedding dimension of 128, two layers, four attention heads, $\alpha = 0.2$, a learning rate of 0.001, and a dropout rate of 0.2. For the Amazon dataset, the ideal configuration featured an embedding dimension of 32, two layers, four attention heads, $\alpha = 0.4$, a learning rate of 0.001, and a dropout rate of 0.5. The best setup for Rotten Tomatoes was identical to that of IMDB, with an embedding dimension of 128, two layers, four attention heads, $\alpha = 0.2$, a learning rate of 0.001, and a dropout rate of 0.2. All configurations utilized `concat=False`.

For the baseline methods, KANN [15], we followed the hyperparameter configurations suggested in their original papers. The model weights were optimized using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 1.0 \times 10^{-9}$. The learning rate was set as 0.0001, and the batch size was fixed at 128. Both the knowledge embedding and latent embedding dimensions were set to 50, resulting in a total dimension $d = 100$ after concatenating the context entity with the entity itself. All hidden layers were configured with a size of 1024, and the number of attention spaces H was set to 4. To manage input length, we retained the portion of reviews that covered 90% of users and items in terms of length, and 80% in terms of quantity.

For the SVD, we utilized the implementation provided by the Surprise library [11] and conducted a grid search to determine the optimal hyperparameter configuration. The search space included the number of latent factors {32, 64, 128, 256}, number of training epochs {20, 30, 40}, learning rate {0.01, 0.001, 0.005, 0.0001}, and regularization term {0.5, 0.05}. We used the biased SVD variant and selected the best configuration based on validation RMSE. The final model was trained with `SVD(n_factors=256, n_epochs=20, biased=True, lr_all=0.005, reg_all=0.05)`.

For the DeepCoNN [25], we adopted the hyperparameter configuration described in the original paper. The length of each review document was fixed at 1000 tokens. Word embeddings were initialized with a dimension of 64. The convolutional layer contained 100 neurons with a kernel window size of 3. The latent dimension for the final user-item interaction layer was set to 10. We used a learning rate of 0.002, a dropout rate of 0.1, and a batch size of 128 throughout the training process.

5.4 Metrics

Recommender systems are evaluated using both rating accuracy and ranking quality. Offline evaluation based on historical data is commonly used [1, 23]. Standard metrics include Root Mean Squared Error (RMSE) for prediction error, and Normalized Discounted Cumulative Gain (nDCG), Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Recall for ranking performance. To assess aspects beyond accuracy, Novelty and Serendipity are also considered. Below, we define each metric used in this study.

RMSE quantifies the deviation between predicted and true ratings:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{r}_i - r_i)^2} \tag{8}$$

MAP combines ranking and relevance:

$$\text{MAP} = \frac{1}{N}\sum_{i=1}^{N}\text{AP}_i, \qquad \text{AP}_i = \frac{1}{m_i}\sum_{k=1}^{K}\text{P@}k \cdot \text{rel}_k \tag{9}$$

Recall@K is the proportion of relevant items retrieved in the top-K recommendations:

$$\text{Recall@}K = \frac{|\text{Rel}_K|}{|\text{Rel}|}. \tag{10}$$

MRR measures how early the first relevant item appears in the ranked list:

$$\text{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\text{rank}_i}, \tag{11}$$

where rank_i is the position of the first relevant item for user i.

nDCG evaluates ranking quality by emphasizing higher-ranked relevant items:

$$\text{nDCG@}K = \frac{\text{DCG@}K}{\text{IDCG@}K}, \tag{12}$$

$$\text{DCG@}K = \sum_{i=1}^{K}\frac{r_i}{\log_2(i+1)}, \quad \text{IDCG@}K = \sum_{i=1}^{|REL|}\frac{1}{\log_2(i+1)}, \tag{13}$$

where, r_i is the binary relevance score, and $|REL|$ is the number of relevant items up to rank K.

Novelty measures how uncommon recommended items are, based on item popularity in the training data.

$$\text{Novelty@K} = \frac{1}{K}\sum_{i \in R_u^K} -\log_2 P(i). \tag{14}$$

The final novelty score is the average across all recommended items in the list.

Serendipity captures how surprisingly relevant items are, relative to the user's interaction history. It is based on the dissimilarity (1 minus cosine similarity) between a recommended relevant item and the set of previously interacted items:

$$\text{Serendipity@}K = \frac{1}{|\text{Rel}_K|}\sum_{i \in \text{Rel}_K}\Big(1 - \cos(\mathbf{v}_i, \mathbf{v}_{\text{history}})\Big) \tag{15}$$

where $\mathbf{v}_i$ is the embedding of item i, and $\mathbf{v}_{\text{history}}$ is the average embedding of items in the user's history.

5.5 Methodology

We conducted our experiments using a 5-fold cross-validation protocol to ensure robust results. Additionally, we guarantee that every user-item interaction is part of the test set in exactly one fold. For the GRAFT, we implemented an early-stopping mechanism based on the validation set's RMSE to prevent overfitting, with training limited to a maximum of 30 epochs for each dataset. The final reported metrics are the average of the test set results from the best-performing run of each fold, where "best" is defined as the run that achieved the lowest validation RMSE. For fairness and reproducibility, we recorded the data partitioning and maintained consistency across all baseline comparisons. Additionally, to isolate the impact of model architecture, we initialized GRAFT, KANN, and DeepCoNN with identical review embedding representations. We derived these features from the all-mpnet-base-v2 sentence transformer model. We kept these embeddings frozen during training to ensure a fair and direct comparison of the models' capabilities. To assess performance beyond observed interactions, we applied ranking metrics over a recommendation list constructed for each user. This list consisted of the ground-truth items, defined as those rated above 70%, ratings greater than 7.0 for IMDb, and 3.5 for Amazon and Rotten Tomatoes, combined with 100 randomly sampled negative items. We drew these negative items from the global item pool, excluding those the user had previously interacted with. This setup enables evaluation of the model's ability to distinguish relevant items from non-relevant ones in a realistic recommendation scenario.

6 RESULTS AND ANALYSIS

This section presents the results and analysis of GRAFT, which we propose, as well as the baseline models, including the RMSE and ranking metrics. It encompasses an exploratory study of the results of the topic extraction. Additionally, we conduct a qualitative analysis that aims to explore aspects beyond traditional metrics, with a focus on the quality of the rankings that we recommend.

6.1 Topics Extraction

We applied BERTopic to extract latent topics from review datasets, using contextualized embeddings and dimensionality reduction to cluster semantically similar documents. We characterized each topic by its most representative terms and associated reviews. To enhance topic quality, we conducted a qualitative inspection of the extracted topics. We found that certain high-frequency but semantically weak terms (e.g., "movie," "series," "watching") reduced the distinctiveness of the topics. As a result, we refined the CountVectorizer step by extending the stopword list to exclude such domain-specific, non-informative words, leading to more coherent and meaningful topic representations.

We derived the extracted topics from frequent word groupings and reflect recurring themes and sentiments across the reviews. Through a curated process of analysis, we observed that many topics capture narrative elements such as "story line" and "character development," expressions of evaluation including dissatisfaction (e.g., "bad quality") or recommendation (e.g., "recommend," "worth"), as well as associations with specific genres and cultural references (e.g., "documentary," "Downton Abbey," "Star Wars," "Pixar").

For the **Amazon** dataset, topics revealed both positive and negative sentiment toward content and quality. Topic 0 captures dissatisfaction with plot and character development, while Topic 1 centers on complaints about poor quality and overall disappointment. Topics 2 and 3 highlight user opinions about comedic elements and characters. Topics 4 and 5 focus on special effects, adaptations, and mixed responses to storylines. Topic 6 reflects positive opinions on documentaries and informative material, whereas Topics 7 and 8 emphasize entertaining storylines and favorite selections. Topic 9 highlights praise for the *Downton Abbey* series, and Topic 10 highlights references to science fiction franchises such as *Terminator* and *Star Wars*.

In the **IMDb** dataset, topics are more genre-driven. Topic 0 reflects appreciation for compelling storytelling. Topics 1 and 2 capture emotional connections with themes of love, music, and superhero characters, respectively. Topics 3 and 4 represent an interest in action-oriented content, with Topic 4 specifically aligning with the style of Tarantino. Topic 5 covers animated Disney films, while Topic 6 focuses on war narratives. Topic 7 reflects interest in horror, especially zombie-related content. Topics 8 and 9 express general praise for well-told stories and fan-favorite franchises such as *Harry Potter* and *Pirates of the Caribbean*.

The **Rotten Tomatoes** dataset presents a mix of technical and emotional feedback. Topic 0 discusses characters, plot, and the structure of action. Topic 1 highlights directorial quality, performance, and soundtrack. Topics 2 and 7 reflect entertainment value and comedic content, while Topics 3 and 4 focus on documentaries and strong acting performances. Topic 5 is associated with horror and thriller genres, and Topic 6 revolves around romantic and comedic themes. Topic 8 includes family-friendly animated content, particularly Disney-related films. Topics 9 and 10 refer to sequels, remakes, and nostalgic action icons like James Bond and Clint Eastwood. Lastly, Topic 11 includes Spanish-language reviews, some of which reference director Francis Ford Coppola.

6.2 Recommendation Models

Table 2 presents the performance comparison across the baselines on the datasets, where the best results are in bold. The proposed GRAFT algorithm demonstrates competitive and stable performance, especially in terms of rating prediction (RMSE).

On the Amazon dataset, GRAFT achieves the lowest RMSE (1.0413), indicating superior accuracy in rating prediction compared to SVD (1.0525), DeepCoNN (1.0792), and KANN (1.2297). However, SVD outperforms GRAFT in all ranking-based metrics (nDCG@10, Recall@10, MRR@10, and MAP@10), suggesting that traditional latent factor models may still be more effective at producing top-ranked item lists under sparse conditions. For the IMDb dataset, GRAFT again records the lowest RMSE (2.0337), demonstrating its robustness in rating prediction even under increased sparsity. Nonetheless, SVD leads in all ranking metrics here as well. GRAFT remains competitive in MRR and MAP, demonstrating that while it may not produce the most relevant items as effectively as SVD, it consistently learns to rank moderately relevant items. On the Rotten Tomatoes dataset, GRAFT achieves the best RMSE (0.7652), reaffirming its strength in accurate rating prediction. In contrast, SVD again outperforms all models in nDCG@10, Recall@10, MRR@10,

Table 2: Performance on Amazon Movies & TV, reported as mean ± standard deviation (SD) over 5-fold cross-validation.

Algorithm	Dataset	RMSE	nDCG@10	Recall@10	MRR@10	MAP@10
GRAFT	Amazon Movies and TV	**1.0413±0.0312**	0.0400±0.0014	0.0884±0.0019	0.0256±0.0012	0.0256±0.0012
SVD	Amazon Movies and TV	1.0525±0.0315	**0.0663±0.0056**	**0.1475±0.0115**	**0.0418±0.0038**	**0.0419±0.0038**
DeepCoNN	Amazon Movies and TV	1.0792±0.0317	0.0444±0.0174	0.0951±0.0353	0.0289±0.0119	0.0289±0.0119
KANN	Amazon Movies and TV	1.2297±0.0338	0.0491±0.0028	0.1069±0.0057	0.0317±0.0020	0.0317±0.0020
GRAFT	IMDb	**2.0337±0.0354**	0.0606±0.0070	0.1189±0.0110	0.0377±0.0046	0.0383±0.0047
SVD	IMDb	2.0451±0.0391	**0.0797±0.0031**	**0.1511±0.0054**	**0.0520±0.0023**	**0.0527±0.0024**
DeepCoNN	IMDb	2.0712±0.0344	0.0405±0.0032	0.0844±0.0067	0.0235±0.0017	0.0239±0.0017
KANN	IMDb	2.4935±0.0560	0.0447±0.0059	0.0908±0.0127	0.0272±0.0039	0.0276±0.0039
GRAFT	Rotten Tomatoes	**0.7652±0.0084**	0.0894±0.0041	0.2026±0.0074	0.0555±0.0039	0.0556±0.0039
SVD	Rotten Tomatoes	0.7770±0.0086	**0.1457±0.0051**	**0.3017±0.0080**	**0.0980±0.0054**	**0.0981±0.0053**
DeepCoNN	Rotten Tomatoes	0.7789±0.0097	0.0295±0.0024	0.0645±0.0037	0.0190±0.0020	0.0190±0.0020
KANN	Rotten Tomatoes	1.0161±0.0074	0.0478±0.0098	0.1022±0.0171	0.0311±0.0074	0.0312±0.0074

and MAP@10. This consistent gap in ranking metrics suggests that GRAFT's current architecture favors precision in rating regression over high-ranking item retrieval.

Paired t-tests were conducted to evaluate the performance of the proposed GRAFT model against the strongest baseline, SVD, using RMSE and ranking-based metrics (nDCG@10, Recall@10, MRR@10, and MAP@10). Results indicate that GRAFT significantly outperformed SVD in terms of RMSE across all datasets: *Amazon* ($t(4) = -10.24$, $p = 0.0005$), *IMDb* ($t(4) = -3.14$, $p = 0.0349$), and *Rotten Tomatoes* ($t(4) = -3.78$, $p = 0.0194$), confirming its advantage in rating prediction accuracy.

Despite SVD obtaining higher absolute scores on ranking metrics, GRAFT consistently showed statistically significant differences in all cases. For nDCG@10, GRAFT was significantly lower than SVD on *Amazon* ($t(4) = -11.88$, $p = 0.0003$), *IMDb* ($t(4) = -6.88$, $p = 0.0023$), and *Rotten Tomatoes* ($t(4) = -14.81$, $p = 0.0001$). The same trend held for Recall@10 (*Amazon*: $t(4) = -12.30$, $p = 0.0003$; *IMDb*: $t(4) = -8.50$, $p = 0.0011$; *Rotten Tomatoes*: $t(4) = -15.65$, $p = 0.0001$), MRR@10 (*Amazon*: $t(4) = -11.37$, $p = 0.0003$; *IMDb*: $t(4) = -6.83$, $p = 0.0024$; *Rotten Tomatoes*: $t(4) = -12.53$, $p = 0.0002$), and MAP@10 (*Amazon*: $t(4) = -11.38$, $p = 0.0003$; *IMDb*: $t(4) = -6.82$, $p = 0.0024$; *Rotten Tomatoes*: $t(4) = -12.65$, $p = 0.0002$).

Table 3 presents the novelty and serendipity metrics for the GRAFT, SVD, DeepCoNN, and KANN models across the datasets. GRAFT demonstrates a consistently balanced performance, achieving high novelty scores (e.g., 15.03 ± 0.03 on *Amazon*) and moderate but stable serendipity, particularly excelling on *Rotten Tomatoes* (0.1818 ± 0.0073). This indicates that GRAFT is capable of offering diverse recommendations while maintaining relevance and surprise, aligning well with user interests.

Table 3 presents the novelty and serendipity metrics for the GRAFT, SVD, DeepCoNN, and KANN models across the *Amazon*, *IMDb*, and *Rotten Tomatoes* datasets. GRAFT demonstrates a consistent and balanced behavior, achieving moderate novelty scores (e.g., 15.03 ± 0.03 on *Amazon*, which are slightly lower than other models) but notably higher serendipity than DeepCoNN and KANN, especially on *Rotten Tomatoes* (0.1818 ± 0.0073). This indicates GRAFT's strength in recommending items that are not necessarily the most obscure but are still diverse and contextually relevant, often resulting in pleasant user surprise.

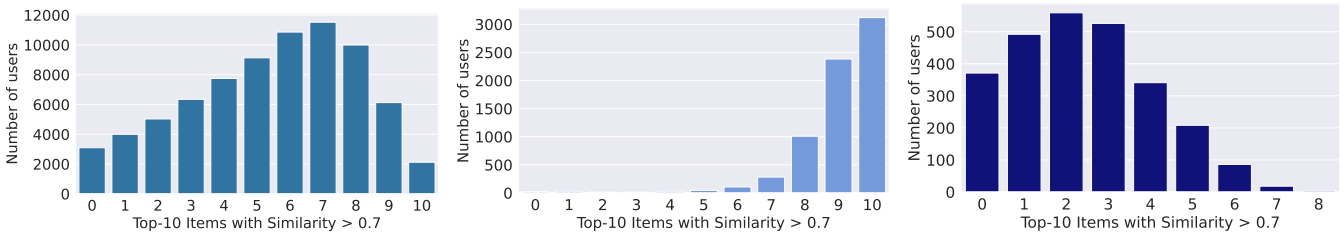
Table 3: Novelty and Serendipity metrics (mean± SD) for each model on the Amazon, IMDb, and Rotten Tomatoes datasets, calculated over 5-fold cross-validation.

Model	Dataset	Novelty	Serendipity
GRAFT	Amazon	15.0317 ± 0.0335	0.0550 ± 0.0009
SVD	Amazon	14.9348 ± 0.0157	**0.1583 ± 0.0091**
DeepCoNN	Amazon	**15.5111 ± 0.1929**	0.0079 ± 0.0069
KANN	Amazon	15.4146 ± 0.0175	0.0309 ± 0.0111
GRAFT	IMDb	9.9088 ± 0.0238	0.0958 ± 0.0086
SVD	IMDb	10.0486 ± 0.0095	**0.1487 ± 0.0040**
DeepCoNN	IMDb	10.0539 ± 0.0520	0.0844 ± 0.0068
KANN	IMDb	**10.5072 ± 0.0014**	0.0085 ± 0.0037
GRAFT	Rotten Tomatoes	13.8079 ± 0.0272	0.1818 ± 0.0073
SVD	Rotten Tomatoes	13.6807 ± 0.0153	**0.2994 ± 0.0126**
DeepCoNN	Rotten Tomatoes	**15.2170 ± 0.0965**	0.0645 ± 0.0038
KANN	Rotten Tomatoes	15.0869 ± 0.0021	0.0195 ± 0.0122

SVD stands out with the highest serendipity values across all datasets, reaching 0.2988 ± 0.0081 on *Rotten Tomatoes*. Still, it does so at the expense of novelty, especially on *IMDb*, where its novelty drops to 10.51 ± 0.0026. This suggests that SVD favors more conventional or popular items that may occasionally be unexpected but are less exploratory.

DeepCoNN, in contrast, demonstrates the highest novelty values in most scenarios (e.g., 15.51 ± 0.19 on *Amazon*) but suffers from extremely low serendipity on that same dataset (0.0079 ± 0.0069). This imbalance suggests that DeepCoNN often recommends rare or lesser-known items, but these may not effectively surprise or satisfy the user, indicating weaker personalization.

KANN performs close to SVD in novelty (e.g., 15.41±0.02 on *Amazon*). Still, its serendipity is substantially lower across all datasets, particularly on *IMDb* (0.0085 ± 0.0037) and *Rotten Tomatoes* (0.0195 ± 0.0122). This suggests that while KANN is capable of suggesting



(a) Amazon Movies and TV **(b) IMDb** **(c) Rotten Tomatoes**

Figure 1: Distribution of high-similarity items (>0.7) in top-10 ranked recommendations across datasets. Each subplot shows the number of users who received a given count of high-similarity items in their recommendation list.

novel items, it struggles to make those recommendations contextually surprising or relevant to users.

6.3 Discussion and the GRAFT Analysis

These results confirm that although GRAFT does not achieve the best top-k ranking metrics, the differences with SVD are statistically significant. The consistent improvement in RMSE, paired with competitive performance in ranking metrics, underscores GRAFT's unique capacity to capture nuanced user-item relationships through review-driven graph modeling, even if those advantages do not directly translate into higher rank-based scores. Overall, GRAFT demonstrates clear advantages in rating accuracy across all datasets. While it does not surpass SVD in ranking metrics, its strength lies in combining review-derived semantic signals with structural graph information to more accurately estimate rating values. This balance underscores the model's effectiveness in learning user and item representations from sparse, semantically rich data.

Nonetheless, this design centers on direct user–item interactions and may underutilize richer, higher-order graph structures that contribute to ranking efficacy. While the gated attention mechanism effectively integrates review semantics to refine user and item representations, yielding superior accuracy in rating prediction, it primarily operates within explicit user–item relations. This direct modeling can limit the propagation of signals across the graph, thereby reducing the model's ability to infer transitive preferences, which are crucial for ranking tasks. Moreover, the fusion gate may attenuate weaker collaborative signals in favor of stronger textual cues, which benefits regression objectives but can hinder relative ranking performance. Overall, the results highlight a trade-off between semantic precision and generalization in ranking, which is intrinsic to GRAFT's design.

To complement the quantitative evaluation of the recommendation model, we implement a qualitative analysis that aims to interpret the model's decision-making process. By examining the alignment between predicted item rankings and ground-truth relevance, we strive to understand how well the model captures meaningful relationships beyond conventional metrics. The analysis evaluates the distribution and proportion of high-similarity recommendations (>0.7) within the top-10 ranked items across the datasets. The goal is to evaluate the model's ability to rank highly similar items within the recommendation list consistently.

The Figure 1 presents a general histogram showing the similarity between the ground truth list and the top-10 ranked recommendation list for each dataset. In Amazon, 54.21% of the recommended items exhibit similarity above the threshold. Moreover, 84.04% of users receive at least three items above this threshold. The Figure 1a reflects this distribution, with a concentration of users receiving between 5 and 9 high-similarity items. For IMDb, surprisingly, the behavior is even more pronounced in contrast to the general results from the models, where this dataset presents the most challenging ratings to predict. The overall proportion of high-similarity items reaches 90.31%, and 99.01% of users are served with three or more such items. Figure 1b confirms this, showing that most users receive between 8 and 10 high-similarity items. Rotten Tomatoes presents a contrasting behavior. Only 24.05% of the top-10 items exceed the similarity threshold, and just 45.41% of users receive three or more despite the similarity proportions being relatively balanced across the top-K positions (ranging from 21.96% to 25.45%). Figure 1c shows a marked decline in the number of users with more than 7 high-similarity items. The absence of values for ranks 9 and 10 is not a visualization artifact, but rather reflects that no users received 9 or more items with similarity above 0.7. Nevertheless, the corresponding proportions in top-9 and top-10 indicate that these ranks still contain relevant items across users, although not concentrated within individual recommendation lists.

Overall, the model demonstrates strong alignment in the IMDb dataset, moderate performance in Amazon, and lower alignment in Rotten Tomatoes. These trends highlight the variation in user-item similarity structure across datasets and emphasize the importance of analyzing both visual and quantitative aspects when evaluating recommender system behavior.

6.4 Limitations and Points of Improvement

This work presents certain limitations that define the scope of the current proposal. First, the model does not account for temporal aspects. All reviews are treated equally, regardless of when they were written, even though more recent reviews often carry more weight in influencing user behavior and item perception. Second, like many text-based recommenders, GRAFT relies on a sufficient volume of explicit reviews to extract reliable topics and gating signals; in cold-start scenarios, where new users or items have few or no reviews, its performance may degrade. Lastly, while the use of topic information allows for some level of semantic abstraction,

the model does not incorporate sentiment analysis in the review processing pipeline. As a result, explicit emotional tone or polarity expressed in the text is not directly captured or modeled. These aspects define the current boundaries of the approach and indicate areas where the methodology could be strengthened.

7 CONCLUSION AND FUTURE REMARKS

In this work, we proposed GRAFT: a Gated Review Attention Framework for Topics in Graph-Based Recommenders. Our approach integrates topic representations extracted from user reviews into a graph attention architecture, enabling more nuanced modeling of user preferences and item characteristics. By incorporating a gating mechanism inspired by recurrent neural networks, the model learns to dynamically control the influence of review-derived topics alongside traditional latent embeddings. Experimental results across multiple datasets demonstrate that GRAFT consistently enhances predictive performance and recommendation quality, particularly when textual context contributes significantly to item differentiation. These findings reinforce our hypothesis that incorporating semantic information from reviews enhances the model's ability to generate more expressive user and item representations. Both quantitative results and similarity-based qualitative analysis confirm that the model retrieves contextually coherent recommendations aligned with user preferences.

Despite these promising results, several limitations point to future directions. First, temporal aspects of reviews, such as recency and evolving user interests, were not considered, although they may affect the weight or relevance of extracted topics. Second, the model's attention and gating layers increase dimensionality and computational cost in proportion to the number of attention heads and layers, raising concerns about scalability. Efficient neighbor sampling strategies and dimensionality reduction techniques could help mitigate this. Moreover, adapting the model for constrained or edge environments remains an open challenge that necessitates further exploration.

ACKNOWLEDGMENTS

We want to thank the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES – Brazil) for funding this research under Grant Code 001. This research was also supported in part by the FAPESB INCITE PIE0002/2022 grant and FAPESB PPF0001/2021grant.

REFERENCES

[1] Charu C. Aggarwal. 2016. *Recommender Systems: The Textbook* (1st ed.). Springer Publishing Company, Incorporated.

[2] Pablo Castells and Dietmar Jannach. 2023. Recommender Systems: A Primer. arXiv:2302.02579 [cs.IR]

[3] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural Attentional Rating Regression with Review-level Explanations. In *Proceedings of the 2018 World Wide Web Conference* (Lyon, France) *(WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1583–1592. https://doi.org/10.1145/3178876.3186070

[4] Li Chen, Guanliang Chen, and Feng Wang. 2015. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction* 25, 2 (June 2015), 99–154. https://doi.org/10.1007/s11257-015-9155-5

[5] Tao Chen, Premaratne Samaranayake, XiongYing Cen, Meng Qi, and Yi-Chen Lan. 2022. The Impact of Online Reviews on Consumers' Purchasing Decisions: Evidence From an Eye-Tracking Study. *Frontiers in Psychology* Volume 13 - 2022 (2022). https://doi.org/10.3389/fpsyg.2022.865702

[6] Zhiyong Cheng, Ying Ding, Xiangnan He, Lei Zhu, Xuemeng Song, and Mohan Kankanhalli. 2018. A3NCF: an adaptive aspect attention model for rating prediction. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (Stockholm, Sweden) *(IJCAI'18)*. AAAI Press, 3748–3754.

[7] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. arXiv:1406.1078 [cs.CL] https://arxiv.org/abs/1406.1078

[8] Shivangi Gheewala, Shuxiang Xu, Soonja Yeom, and Sumbal Maqsood. 2024. Exploiting deep transformer models in textual review based recommender systems. *Expert Systems with Applications* 235 (2024), 121120. https://doi.org/10.1016/j.eswa.2023.121120

[9] Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv:2203.05794 [cs.CL] https://arxiv.org/abs/2203.05794

[10] Xiangfu He, Qiyao Peng, Minglai Shao, and Yueheng Sun. 2024. Diffusion Review-Based Recommendation. In *Knowledge Science, Engineering and Management*, Cungeng Cao, Huajun Chen, Liang Zhao, Junaid Arshad, Taufiq Asyhari, and Yonghao Wang (Eds.). Springer Nature Singapore, Singapore, 255–269.

[11] Nicolas Hug. 2020. Surprise: A Python library for recommender systems. *Journal of Open Source Software* 5, 52 (2020), 2174. https://doi.org/10.21105/joss.02174

[12] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. 2010. *Recommender Systems: An Introduction* (1st ed.). Cambridge University Press, USA.

[13] N. Zafar Ali Khan and R. Mahalakshmi. 2023. A novel user review-based contextual recommender system. *International Journal of Modeling, Simulation, and Scientific Computing* 14, 01 (2023), 2341002. https://doi.org/10.1142/S1793962323410027 arXiv:https://doi.org/10.1142/S1793962323410027

[14] Zheng Li, Di Jin, and Ke Yuan. 2023. Attentional factorization machine with review-based user–item interaction for recommendation. *Scientific Reports* 13, 1 (2023), 1–17. https://doi.org/10.1038/s41598-023-40633-4

[15] Yun Liu and Jun Miyazaki. 2022. Knowledge-aware attentional neural network for review-based movie recommendation with explanations. *Neural Comput. Appl.* 35, 3 (sep 2022), 2717–2735. https://doi.org/10.1007/s00521-022-07689-1

[16] Cataldo Musto, Marco de Gemmis, Giovanni Semeraro, and Pasquale Lops. 2017. A Multi-criteria Recommender System Exploiting Aspect-based Sentiment Analysis of Users' Reviews. In *Proceedings of the Eleventh ACM Conference on Recommender Systems* (Como, Italy) *(RecSys '17)*. Association for Computing Machinery, New York, NY, USA, 321–325. https://doi.org/10.1145/3109859.3109905

[17] Pedro Pires, Bruno Rizzi, and Tiago Almeida. 2024. Why Ignore Content? A Guideline for Intrinsic Evaluation of Item Embeddings for Collaborative Filtering. In *Proceedings of the 30th Brazilian Symposium on Multimedia and the Web* (Juiz de Fora/MG). SBC, Porto Alegre, RS, Brasil, 345–354. https://doi.org/10.5753/webmedia.2024.243199

[18] Shaina Raza and Chen Ding. 2022. News recommender system: a review of recent progress, challenges, and opportunities. *Artif. Intell. Rev.* 55, 1 (Jan. 2022), 749–800. https://doi.org/10.1007/s10462-021-10043-x

[19] Fu Shang, Jiatu Shi, Yadong Shi, and Shuwen Zhou. 2024. Enhancing E-Commerce Recommendation Systems with Deep Learning-based Sentiment Analysis of User Reviews. *International Journal of Engineering and Management Research* 14, 4 (aug 2024). https://doi.org/10.5281/zenodo.13221409

[20] Petar Veličković, Arantxa Casanova, Pietro Liò, Guillem Cucurull, Adriana Romero, and Yoshua Bengio. 2018. Graph attention networks. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* (2018), 1–12. https://doi.org/10.1007/978-3-031-01587-8_7 arXiv:1710.10903

[21] Qiang Wang, Wen Zhang, Jian Li, Feng Mai, and Zhenzhong Ma. 2022. Effect of online review sentiment on product sales: The moderating role of review credibility perception. *Computers in Human Behavior* 133 (2022), 107272. https://doi.org/10.1016/j.chb.2022.107272

[22] Shuang Yang and Xuesong Cai. 2023. An Enhanced Recommendation Model Based on Review Text Graph and Interaction Graph. *IEEE Access* 11 (2023), 88234–88244. https://doi.org/10.1109/ACCESS.2023.3305954

[23] Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. *ACM Comput. Surv.* 55, 8, Article 170 (Dec. 2022), 38 pages. https://doi.org/10.1145/3556536

[24] André Zanon, Leonardo Rocha, and Marcelo Manzato. 2024. O Impacto de Estratégias de Embeddings de Grafos na Explicabilidade de Sistemas de Recomendação. In *Proceedings of the 30th Brazilian Symposium on Multimedia and the Web* (Juiz de Fora/MG). SBC, Porto Alegre, RS, Brasil, 231–239. https://doi.org/10.5753/webmedia.2024.241857

[25] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint Deep Modeling of Users and Items Using Reviews for Recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* (Cambridge, United Kingdom) *(WSDM '17)*. Association for Computing Machinery, New York, NY, USA, 425–434. https://doi.org/10.1145/3018661.3018665

[26] Yuanyuan Zhuang and Jaekyeong Kim. 2021. A BERT-Based Multi-Criteria Recommender System for Hotel Promotion Management. *Sustainability* 13, 14 (2021). https://doi.org/10.3390/su13148039