Graph-based Extractive Explainer for Recommendations

Peng Wang*
University of Virginia
Charlottesville, VA, USA
pw7nc@virginia.edu

Renqin Cai*
University of Virginia
Charlottesville, VA, USA
rc7ne@virginia.edu

Hongning Wang
University of Virginia
Charlottesville, VA, USA
hw5x@virginia.edu

ABSTRACT

Explanations in a recommender system assist users make informed decisions among a set of recommended items. Extensive research attention has been devoted to generate natural language explanations to depict how the recommendations are generated and why the users should pay attention to them. However, due to different limitations of those solutions, e.g., template-based or generation-based, it is hard to make the explanations easily perceivable, reliable, and personalized at the same time.

In this work, we develop a graph attentive neural network model that seamlessly integrates user, item, attributes and sentences for extraction-based explanation. The attributes of items are selected as the intermediary to facilitate message passing for user-item specific evaluation of sentence relevance. And to balance individual sentence relevance, overall attribute coverage and content redundancy, we solve an integer linear programming problem to make the final selection of sentences. Extensive empirical evaluations against a set of state-of-the-art baseline methods on two benchmark review datasets demonstrated the generation quality of proposed solution.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; **Summarization**; **Personalization**; • **Computing methodologies** → **Multi-task learning**; **Neural networks**.

KEYWORDS

Extraction-based explanation, graph neural networks

ACM Reference Format:
Peng Wang, Renqin Cai, and Hongning Wang. 2022. Graph-based Extractive Explainer for Recommendations. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3485447.3512168

1 INTRODUCTION

Nowadays, recommendations in online information service platforms, from e-commerce (such as Amazon and eBay) to streaming services (such as Netflix and youtube), have greatly shaped everyone's life, by affecting who sees what and when [1, 11, 21]. Therefore, besides improving the quality of recommendations, explaining the recommendations to the end users, e.g., how the recommendations are generated [6, 28, 34, 40] and why they should pay attention to the recommended content [37, 43, 47], is also critical to improve users' engagement and trust in the systems [2, 12, 30].

To be helpful in users' decision making, the system-provided explanations have to be easily perceivable, reliable, and personalized. Template-based explanations have been the dominating choice, where various solutions were developed to extract attribute keywords or attribute-opinion pairs from user reviews [5, 34, 37, 43, 47] or from pre-existing knowledge graph [40, 42] to form the explanation content about a specific item for a target user. The fidelity of the generated explanations can be improved by careful quality control in the algorithms' input. But the predefined templates lack desirable diversity, and their rigid format and robotic style are less appealing to ordinary users [16, 45].

On the other hand, due to the encouraging expressiveness of the content generated from neural language models [3, 10, 26], an increasing number of solutions adopt generative models for explanation generation [16, 17, 45]. The generated content from such solutions are generally believed to have better readability and variability. Nevertheless, the high complexity of neural language models prevents fine-grained control in its generated content. And the success of such models heavily depends on the availability of large-scale training data. Due to the lack of observations about opinionated content from individual users on specific items, the generation from such models can hardly be personalized. On the contrary, it has been observed that such models' output tend to be generic and sometimes even less relevant to target items [45].

To understand the aforementioned advantages and limitations of these two types of explanation generation methods, we extract a few sample outputs from one typical solution of each type trained on the same hotel recommendation dataset in Table 1. We chose Explicit Factor Model (EFM) [47] to represent template-based solutions, and Sentiment Aligned Explainable Recommendation (SAER) [45] to represent neural generative solutions. The robotic style of EFM's explanation content can be easily recognized, e.g., only the attribute keyword changes across its output for different items. Even if the recommended hotels in the example are indeed featured with *staff* or *location*, such generic content hurts the trustworthiness of the explanations. On the other hand, although SAER's output style is more diverse, its content is quite generic; especially when comparing across items, the aspects mentioned are less specific about the target items. This is also problematic when a user needs to choose from a set of recommended items based on their explanations.

To make the explanations easily perceivable, reliable, and also personalized, we propose an extractive solution, named GRaph Extractive ExplaiNer (GREENer), to extract sentences from existing reviews for each user-item pair as explanations. By collectively selecting from existing review content, the extracted sentences

*These authors contributed equally to this work

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9096-5/22/04.
https://doi.org/10.1145/3485447.3512168

Table 1: Example explanations produced by 3 different types of explainable recommendation models.

Hotel	Kimpton Hotel Eventi, NYC	New Orleans Marriott
EFM	You might be interested in staff/room, on which this hotel performs well.	You might be interested in staff/location, on which this hotel performs well.
SAER	The room was spacious and the room was clean. I was very pleased with the staff, the hotel and staff were very friendly.	The location was great, the hotel was very nice, and the rooms were clean and comfortable. The room was spacious and the beds were extremely comfortable.
GREENer	The view of the empire state building was incredible! The hotel was beautifully decorated, and the staff was very helpful. The room was huge and very clean, the bed comfy and the jacuzzi wonderful.	There was a great view of the Mississippi river. Can't beat the location, just a few blocks from the heart of bourbon street.

maintain the readability from human-written content, and thus make the explanations easily perceivable. For a given pair of user and item, the past reviews from the user suggest his/her preferences on different aspects/attributes of this type of items; and the past reviews describing the item suggest its commonly discussed aspects. Hence, specificity about the user and item can be captured, which leads to personalized explanations. And because of the aggregation among user-provided content about the item, the reliability of the selected content can also be improved.

Accurate extraction from existing content as explanations is however non-trivial. First, not all the sentences in a user review are relevant to the item. For example, it is very common to encounter users' personal experiences mentioned in a review. Such content is clearly unqualified as explanations and should be filtered. Second, the user and item should play different roles in selecting the sentences for explanation: the item suggests the set of relevant aspects, while the user suggests where the attention should be paid to. Therefore, the interplay between the user and item should be carefully calibrated when evaluating a sentence's relevance. Third, the selected sentences should cover distinct aspects of an item; and it is apparently undesirable to repeatedly mention the same aspect in different sentences when explaining an item. However, it is expected that an item's popular attributes will be mentioned in multiple users' reviews with some content variations. Avoiding such nearly duplicated content becomes necessary and challenging.

To address these challenges in extractive explanation generation, we develop a graph attentive neural network model that seamlessly integrates user, item, attributes and sentences for sentence selection. For a collection of items, we first extract frequently mentioned attributes as the intermediary to connect users and items with sentences, i.e., the connectivity on the graph suggests who mentioned what about the item. As a result, sentences not related to any selected attributes are automatically filtered. To handle data sparsity when estimating the model parameters, we employ pre-trained language models [9, 24] for attribute words' and sentences' initial encoding. Through attentive message passing on the graph, heterogeneous information from user, item and attributes about the candidate sentences is aggregated for user-item specific evaluation of sentence relevance. However, because each sentence is independently evaluated by the neural network, content redundancy across sentences cannot be directly handled. We introduce a post-processing strategy based on Integer Linear Programming to select the final top-K output, where the trade-off between relevance and redundancy is optimized.

To investigate the effectiveness of GREENer for explanation generation, we performed extensive experiments on two large public review datasets, Ratebeer [45] and TripAdvisor [36]. Compared with state-of-the-art solutions for explanations, GREENer improved the explanation quality in both BLEU [23] and ROUGE [18] metrics. Our ablation analysis further demonstrated the importance of modeling these four types of information source for explanation generation, and also the importance of a graph structure for capturing the inter-dependency among them. Our case studies suggest that our produced explanations are more perceivable, specific to the target user-item pair, and thus more reliable.

2 RELATED WORK

Numerous studies have demonstrated that explanations play an important role in helping users evaluate results from a recommender system [4, 7, 31, 32, 41, 46]. And various forms of explanations have been proposed, from social explanations such as "*X, Y and 2 other friends like this.*" [29], to item relation explanations such as "*A and B are usually purchased together.*" [19, 40, 42], and opinionated text explanations such as "*This **phone** is featured with its high-resolution **screen**.*" [37, 45, 47], which is the focus of this work.

There are currently three mainstream solutions to generate opinionated textual explanations, namely template-based, generation-based, and extraction-based methods. They all work on user-provided item reviews to create textual explanations. In particular, template-based methods predict important attributes of the target item together with the sentiment keywords from user reviews to fill in the slots in those manually crafted templates. As typical solutions of this type, EFM [47] and MTER [37] predict important item attributes and corresponding user opinion words for a given recommendation via matrix factorization and tensor factorization. EX^3 extracts key attributes to explain a set of recommendations, based on the idea that the selected attributes should predict users' purchasing behavior of those items [43]. CountER employs counterfactual reasoning to select the important aspects for explanation [33]. The main focus in these template-based methods has been devoted to identify the most important item attributes and user opinion, i.e., to improve reliability and personalization; but its lack of content variability and robotic explanation style make such explanations less appealing to the end users.

To increase content diversity in the provided explanations, neural language models are employed in generation-based methods to synthesize natural language explanations. As an earlier work, NRT models explanation generation and item recommendation in a

shared user-item embedding space, where its predicted recommendation rating is used as part of the initial state for corresponding explanation generation [17]. NETE shared a very similar idea with NRT, but it further confines the generation to cover specific item attributes that are selected by a separated prediction module [16]. SAER constrains the sentiment conveyed in the generated explanation content to be close to the item's recommendation score [45]. However, due to the high complexity of neural language models, it is very hard to control such models' content generation at a fine granularity. As a result, the reliability of its generation is questionable. Furthermore, such methods tend to generate generic content to fit the overall data distribution in a dataset. Hence, on the user side, the explanation is less personalized; and on the item side, the explanation could be even less relevant (e.g., overly generic).

Extraction-based solutions directly select representative sentences from the target item's existing reviews. And our proposed solution falls into this category. NARRE selects the most attentive reviews as the explanation, based on the attention originally learned to enrich the user and item representations for recommendation [8]. CARP uses the capsule network for the same purpose [15]. Wang et al. [39] adopt reinforcement learning to extract the most relevant review text that matches a given recommender system's rating prediction. In nature, extraction-based solutions model the affinity between user-item pairs with sentences, which is very sparse: a user review is typically short and a user typically does not write many reviews. Personalized explanation is hard to achieve in such a scenario. Our solution breaks this limitation by introducing item attributes as an intermediary, which not only alleviate sparsity issue but also improves specificity and reliability of the generated explanations (e.g., the sentences will only cover attributes associated with the target item). We should note that the extraction-based solutions are restricted to an item's existing reviews; for items with limited exposure, e.g., a new item, it is hard to generate informative explanations in general. A very recent work combines extractive and generative methods for explanations [44], which has potential to solve the challenge. We leave this direction as our future work.

3 GRAPH EXTRACTIVE EXPLAINER FOR RECOMMENDATIONS

In this section, we describe our proposed extractive explanation solution GREENer in detail. At the core of GREENer is a graph neural network, which integrates heterogeneous information about a user, a recommended item, and all candidate sentences via attentive message passing. To alleviate the observation sparsity issue at the user-item level, we introduce item attributes as an intermediary to connect user, item and sentences. Finally, the extraction is performed by solving an integer linear programming problem to balance individual sentence relevance, overall coverage, and content diversity in the final selections.

3.1 Problem Setup & Notations

We first define the notations employed in this paper. Denote the candidate recommendation item set as C, the set of users as $\mathcal{U}$, and the vocabulary of text reviews as $\mathcal{V}$. We use $|\cdot|$ to denote the cardinality of a set. We collect all the reviews from $\mathcal{U}$ about C and segment them into sentences. Then we denote $\mathcal{S}_{uc} = \{s^i_{uc}\}_{i=1}^N$ as all the sentences from user u about item c, where each sentence $s_{uc} = \{w_i\}_{i=1}^T$ consists of T words $w \in \mathcal{V}$. We aggregate sentences from user u over all items into a set $\mathcal{S}_u = \{\mathcal{S}_{uc}\}_{c \in C}$; and similarly, we aggregate sentences about item c from all users into $\mathcal{S}_c = \{\mathcal{S}_{uc}\}_{u \in \mathcal{U}}$. The attributes $\mathcal{F}$ are items' popular properties mentioned in the whole review corpus, which are a subset of words in $\mathcal{V}$. For a pair of a user $u \in \mathcal{U}$ and an item $c \in C$, an extractive explanation model is to select K sentences $\{s^i\}_{i=1}^K$ from the union of sentences $\mathcal{S}_u \bigcup \mathcal{S}_c$, such that these K sentences best describe how the item is relevant to the user's preference.

3.2 Neural Graph Model for Sentence Encoding

Measuring the relatedness between a candidate sentence and a target pair of user and item is vital for precise extractive explanation. On a given training corpus, the relatedness can be inferred based on the observed associations between user, item, and sentences in the reviews. For example, how does a user usually describe a type of items; and how is an item typically commented by a group of users. But such observations are expected to be sparse, as a user often provides only a handful of reviews.

GREENer addresses the sparsity issue by introducing item attributes as the intermediary between a user-item pair and associated sentences, and accordingly captures the relatedness from two complementary perspectives. First, the observed sentences written by the user and those describing the item, connected via the attributes appearing in these sentences, suggest the direct dependence between a sentence and the pair of user and item. This co-occurrence relation forms a heterogeneous graph among users, items, attributes, and sentences, suggesting their contextualized dependency. GREENer leverages an attentive graph neural network to model this direct dependence structure. Second, the learnt representations of users, items and sentences also suggest the dependence in the embedding space. GREENer utilizes the feature crossing technique to capture this indirect dependence. In the following, we provide the technical details of these two important components.

3.2.1 Attentive Graph Structure. GREENer utilizes a heterogeneous graph to capture the contextualized dependence among users, items, attributes, and sentences. For each review written by a user about an item, we construct a graph consisting of the user, the item, all sentences written by the user, all sentences describing the item, and the attributes mentioned in these sentences. The detailed graph structure is illustrated in Figure 1. Attributes serve as a bridge to connect the user, item and sentences, via the observed co-occurrences among these four types of entities. Attentive message passing on the graph integrates information from these different types of entities to form their latent representations, which can be used to predict if a candidate sentence is relevant to a given user-item pair, i.e., extract as explanation.

Next, we will zoom into the detailed design of this attentive graph for learning the representations of users, items and sentences.

Nodes. A graph g is created for each user-item pair, which consists of four types of nodes: a user node u, an item node c, $|S_g|$ sentence nodes where $\mathcal{S}_g = \mathcal{S}_u \bigcup \mathcal{S}_c$, and M attribute nodes $\{f^i_g\}_{i=1}^M$ where $M \leq |\mathcal{F}|$ represents the attributes appearing in sentences $\mathcal{S}_g$. Note the set of sentences $\mathcal{S}_{uc} = \{s^i_{uc}\}_{i=1}^N$ written by user u about item c is a subset of $\mathcal{S}_g$. Empirically, sentences in $\mathcal{S}_g$ that do not contain any

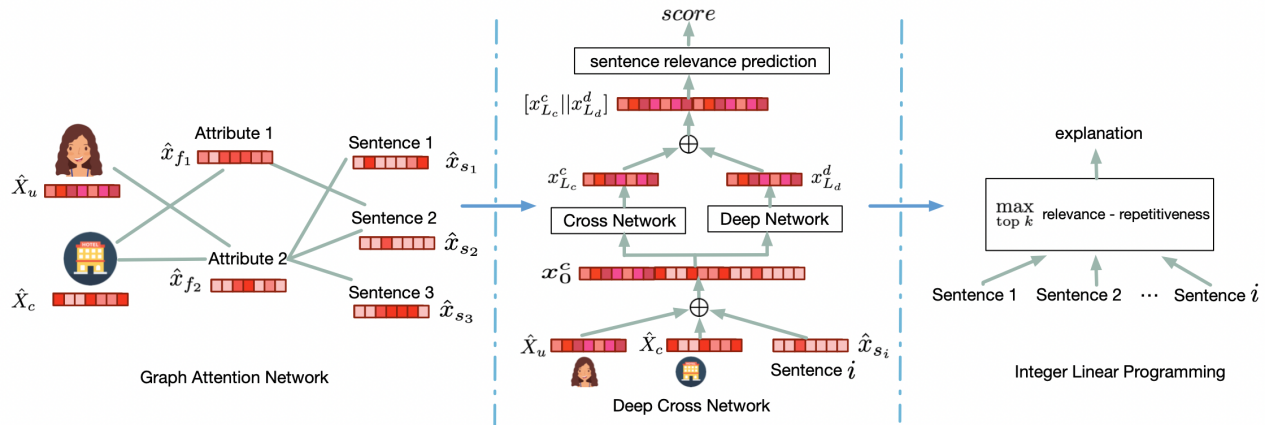

Figure 1: Illustration of GRaph Extractive ExplaiNer (GREENer). For a pair of user and item, GREENer utilizes graph attention network and deep cross network to encode past sentences written by the user and past sentences describing the item. Then it utilizes Integer Linear Programming to select sentences as an explanation.

attribute specific to item c can be filtered to further improve the accuracy of the finally selected sentences.

The input representation of the user node is a dense vector X_u, obtained by mapping the user index u through the input user embeddings $E_u \in \mathbb{R}^{|\mathcal{U}|\times d_u}$. Likewise, the input representation of the item node X_c is obtained in the same manner from $E_c \in \mathbb{R}^{|C|\times d_c}$ accordingly. To obtain good semantic representations, instead of learning sentence representations from scratch, we take advantage of the pre-trained language model BERT [9] to encode sentences $\mathcal{S}_g$ into input node representations. Specifically, we fine-tuned BERT on the review text data in the training set. Then we feed sentences into the fine-tuned BERT to obtain their embedding vectors $\{X_{s_g^i}\}_{i=1}^{|\mathcal{S}_g|}$ as input representations of sentence nodes. Likewise, the input representation of attribute nodes $\{X_{f_g^i}\}_{i=1}^{M}$ is also pre-trained on the review text data using GloVe [24].

Edges. To capture the co-occurrence among different entities in the observed review content, we introduce an edge e_{uf} connecting user node u to attribute node f if the attribute was used by the user in his/her training reviews. Likewise, edge e_{cf} is introduced to connect item node c and attribute node f if the attribute was used to describe the item. Finally, an edge e_{fs} is introduced to connect attribute node f to sentence node s if the sentence contains the attribute word. Notice that all the edges are non-directional by design, as shown in Figure 1. As a result, the attributes serve as a bridge to connect the user and item with individual sentences. For example, a user can now be associated with sentences from other reviews about the item, and so can the item and sentences be. In this work, we only consider the binary edge weight; other type of edge weights, e.g., continuous weights, are left as our future work.

Attentive Aggregation Layer. Given a constructed graph g with nodes $\{X_u, X_c, X_f, X_s\}$ and edges $\{e_{uf}, e_{cf}, e_{fs}\}$, we adopt the attention mechanism from the graph attention networks [35] to encode co-occurrence information into node representations. Specifically, we stack L graph attention layer to map the input node representations into the output node representations $\{\hat{X}_u, \hat{X}_c, \hat{X}_f, \hat{X}_s\}$. Due to the recursive nature of graph attention, we only use one layer as an example to illustrate the design in our solution. For example, in the l-th layer, the inputs to the graph attention layer are $H^l = \{H_u, H_c, H_f, H_s\}$, which correspond to hidden representations of user node, item node, attribute nodes and sentence nodes obtained from the $(l-1)$-th layer. For the i-th node h_i^l in the graph, we obtain attention weights α^l for its connected nodes as,

$$\alpha_{ij}^l = \frac{\exp(z_{ij}^l)}{\sum_{j' \in \mathcal{N}_i} \exp(z_{ij'}^l)}$$

$$z_{ij}^l = \text{LeakyReLU}\big(W_a^l [W_q^l h_i^l || W_k^l h_j^l]\big)$$

where $\mathcal{N}_i$ refers to neighborhood of node i. $\{W_a^l, W_q^l, W_k^l\}$ are parameters to be estimated and $||$ denotes the concatenation operation.

With the attention weights, we obtain the output hidden representation of node i in the l-th layer as

$$h_i^{l+1} = \sigma\Big(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^l h_j^l\Big) \tag{1}$$

With the d_h multi-head attention, we repeat the above process d_h times and merge the output hidden representations from d_h heads as the representation of node i as $h_i^{l+1} = ||_{head=1}^{d_h} h_{head,i}^{l+1}$, where $h_{head,i}^{l+1}$ is obtained by Eq. (1).

Note that for the initial attention layer, we use the input node representations $\{X_u, X_c, X_f, X_s\}$ as the input H^0, and through L attention layers, we use the output representations H^L as the output node representations $\{\hat{X}_u, \hat{X}_c, \hat{X}_f, \hat{X}_s\}$.

3.2.2 Feature Crossing. The attentive graph structure captures the direct co-occurrence dependency between a user-item pair and sentences. From a distinct perspective, feature crossing models the indirect dependency among them on top of the graph representations. Following the design of Deep & Cross network (DCN) [38], GREENer applies feature crossing to model the representation-level interaction among $\hat{X}_u$, $\hat{X}_c$, and $\hat{X}_s$. DCN is a combination of cross network and deep network in a parallel structure as shown in Figure 1. The cross network is a stack of multiple cross layers, which

can be written as

$$x_{l+1}^c = x_0^c x_l^{c\mathsf{T}} w_l^c + b_l^c + x_l^c$$

where x_l^c represents the hidden state in l-th layer of the cross network. The deep network is a multiple layer fully-connected neural network, which can be written as

$$x_{l+1}^d = f(W_l^d x_l^d + b_l^d)$$

where x_l^d represents the hidden state in l-th layer of the deep network. $\{w_l^c, b_l^c, W_l^d, b_l^d\}$ are trainable parameters. These two networks take the same input, which is a concatenation of user, item and sentence output node representation from the group attention layers, as

$$x_0^c = x_0^d = [\hat{X}_u || \hat{X}_c || \hat{x}_{s_i}]$$

Through this DCN module, we can obtain the final output representation sentence s in graph g as

$$x_s^{cd} = [x_{L_c}^c || x_{L_d}^d]$$

where $x_{L_c}^c$ and $x_{L_d}^d$ are the outputs from the cross network and deep network of sentence s, respectively. It aggregates information about a sentence's relatedness to a user-item pair from their direct co-occurrence relation and indirect representation-level dependency.

3.3 Sentence Extraction

Based on the encoded sentence representations, GREENer learns to predict if the sentences are qualified explanations. In addition, GREENer also predicts if an attribute should be mentioned in the extracted explanations, which forms a multi-task objective for parameter estimation.

Multi-Task Objective. In a given training corpus of review sentences, GREENer is trained to rank sentences mostly related to the ground-truth sentences from user u's review about item c (i.e., $\mathcal{S}_{uc}$) above all other sentences. This is realized via a pairwise rank loss function. Specifically, the ranking score $g(s_i)$ of sentence s_i is obtained by its feature crossing representation x_i^{cd} via a linear mapping $g(s_i) = \langle W_o^s, x_i^{cd} \rangle$.

The relevance of a candidate sentence s_i against $\mathcal{S}_{uc}$ can be simply realized via an indicator function, i.e., whether s_i is in $\mathcal{S}_{uc}$. To relax this, we choose to measure the similarity r_i between s_i and $\mathcal{S}_{uc}$,

$$r_i = \max_{s_{uc}^j \in \mathcal{S}_{uc}} sim(s_i, s_{uc}^j) \tag{2}$$

where $sim(\cdot)$ can be any text similarity metric that measures the semantic similarity between a pair of sentences. In our experiments, we used BLEU score, such that sentences in $\mathcal{S}_{uc}$ always have the highest similarity.

Under this similarity-based notion of sentence relevance, the pairwise ranking loss on a set of candidate sentences can be computed as follows,

$$L_s = -\sum_{s_i, s_j \in S_g} \text{sign}(r_i - r_j) \log\ \sigma\big(g(s_i) - g(s_j)\big)$$

where $\sigma(\cdot)$ is the sigmoid function.

In addition to recognizing the qualification of individual sentences, we believe good explanations should also cover important attributes for each user-item pair. This can be achieved by requiring the learnt attribute representations to be predictive about the ground-truth attributes. As a result, we introduce a logistic regression classifier based on the output representation $\hat{x}_{f_i}$ for each attribute node f_i, $p(f_i) = \sigma\big(\langle W_o^f, \hat{x}_{f_i} \rangle\big)$, to predict if f_i should appear in the explanation. We adopt the cross entropy as the loss function of attribute predictions. With the ground-truth label y_{f_i}, i.e., those appear in the ground-truth review content for a user-item pair, the loss of attribute predictions is,

$$L_f = -\sum_{i=1}^{M} y_{f_i} \log p(f_i)$$

Combining these two losses, we obtain the objective function as

$$L = \lambda L_s + (1 - \lambda) L_f$$

where λ is a hyper-parameter to control the weight of each loss to the objective.

Collective Sentence Selection. To reduce redundancy and increase coverage in the extracted sentences, we should select the sentences that are dissimilar to each other but also highly relevant to the target user-item pair. The scoring function $g(s)$ is trained to optimize the latter, but it alone cannot handle the former, which is a combinatorial optimization problem. To find the final K sentences, we adopt Integer Linear Programming (ILP) to solve this optimization problem, which is formulated as follows,

$$\begin{aligned}
\max\ & \sum_{i=1}^{|S_g|} x_{s_i} g(s_i) - \alpha \sum_{i \neq j} y_{ij} sim(s_i, s_j) \\
s.t.\ & \sum_{i=1}^{|S_g|} x_{s_i} = K \\
& x_{s_i} \in \{0, 1\},\ \forall\ i, \\
& y_{ij} \in \{0, 1\},\ \forall\ \{i, j\} \\
& x_{s_i} + x_{s_j} <= y_{ij} + 1 \\
& \sum_{i \neq j} y_{ij} = K * (K - 1)
\end{aligned}$$

where α balances between relevance and content redundancy. As the selection of sentences for each user-item pair can be performed independently, this ILP problem can be solved with parallelism.

4 EXPERIMENTS

In this section, we investigate the effectiveness of our proposed solution GREENer in generating explanations for recommended items. We conducted experiments on two large review-based recommendation datasets and compared our model against a set of state-of-the-art baselines to demonstrate its advantages. In addition, we also performed ablation analysis to study the importance of different components in GREENer.

4.1 Experiment Setup

We chose review-based public recommendation datasets Ratebeer [20] and TripAdvisor [36] for our evaluation purpose. Both datasets contain user-provided textual reviews about their opinions towards specific items, including user ID, item ID, review text content and

Table 2: Summary of the processed datasets. Rb stands for Ratebeer and TA stands for TripAdvisor.

	# Users	# Items	# Reviews	# Sentences	# Attributes
Rb	1,664	1,487	109,746	519,353	572
TA	4,948	4,487	159,834	560,367	503

opinion ratings. In the Ratebeer dataset, the ratings fall into the range of $[1, 20]$; and in the TripAdvisor dataset, the rating's range is $[1, 5]$. Since a recommender system would generally recommend items that are attractive to users, we only focused on user-item interactions with positive ratings. In particular, we used Ratebeer reviews with ratings greater than 10 and TripAdvisor reviews greater than 3 to construct the corpus for our experiments. Since GREENer focuses on explanation generation, in the experiments we directly used the observed item in each user-item pair as the recommendation to be explained.

Data Pre-Processing. Review content has been directly used as ground-truth explanations for evaluation in many previous studies [8, 39]. But as suggested in [22], a large portion of sentences in a review describes personal subjective experience, like "*I drank two bottles of this beer*", which does not provide any information about the reason why the user liked this item, and hence they are not qualified as explanations. In contrast, sentences that serve as explanations should describe the important properties of items to help users make informed decisions, like "*taste is of bubble gum and some banana.*" Therefore, we construct the explanation dataset by keeping informative sentences in the experiments. For both datasets, we used the Sentires toolkit [48] to extract attribute words from reviews and manually filter out inappropriate ones based on our domain knowledge. Then, for each review, we kept sentences that cover at least one of the selected attribute words as candidate explanations.

We also filtered inactive users and unpopular items: we kept users who at least have fifteen reviews and items are associated with at least fifteen reviews. We keep the 20,000 most frequent words in our vocabulary and use the "unk" token to represent the rest words. The statistics of the processed datasets are reported in Table 2. We split the dataset into training, validation, and testing dataset according to the ratio 70%:15%:15%.

Baselines. We compare our model with five baselines, covering both generation-based and extraction-based methods, which can produce natural language sentences as explanations :

- **NRT**: Neural Rating and Tips Generation [17], a generation-based solution. It is originally proposed for tip (a short sentence summary) generation, but can be seamlessly adapted to generating explanations. It utilizes an RNN-based neural language model to generate explanations.
- **SAER**: Sentiment Aligned Explainable Recommendation [45]. This is another generation-based solution. It focuses specifically on the sentiment alignment between the recommendation score and generated explanations. It implements a sentiment regularizer and a constrained decoding method to enforce the sentiment alignment in the explanations in both training and inference phases.
- **NARRE**: Neural Attentional Regression model with Review-level Explanations [8]. It is an extraction-based solution. It learns the usefulness of the existing reviews through attention. It selects the review with highest attention score as the explanation.
- **SEER**: Synthesizing Aspect-Driven Recommendation Explanations from Reviews [14]. This is another extraction-based solution. It takes a user's sentiment towards item's aspects as input, which can be obtained from explainable recommendation models such as EFM [47], and then form an ILP problem to select the K most representative and coherent sentences that fit the user's demand of K aspects.
- **ESCOFILT**: Unsupervised Extractive Summarization-Based Representations for Accurate and Explainable Collaborative Filtering[25]. This is also an extraction-based solution. It is built on BERT text representation to generate user and item profile by clustering user-side and item-side sentence embeddings using K-Means. The K sentences that are the closest to their own cluster centroids are selected to form the explanations.

Implementation Details. For both datasets, we first pre-trained 256-dimension Glove embeddings [24] on the whole vocabulary and fine-tuned BERT model on our dataset using Sentence-BERT [27] which were later used to initialize the attribute node and sentence node, respectively. User and item node embedding size d_u and d_c were both set to 256. We stacked 2 GAT layers, with 4 head in the first layer and 1 head in the second layer. The hidden dimension size in GAT was set to 256. For DCN, we combined a 2-layer cross network with a 2-layer MLP whose hidden size and output size were both set to 128. The sentence relevance score r_i in Eq (2) used in pairwise loss was the pre-computed maximum sentence BLEU score between s_i and $\mathcal{S}_{uc}$.

During training, we used a batch size of 16 and applied Adam optimizer [13] with a learning rate of 2e-4. The λ in multi-task loss function was set to 0.5. The model was selected according to its performance on the valid set where we took the top-5 predicted sentences and computed the BLEU score with the ground-truth review. When finally generating the explanations, in order to reduce the computation complexity of the ILP problem, we selected the top-100 predicted sentences for each user-item pair on the test set and use the Gurobi[1] solver to select the top-5 most relevant yet non-repetitive sentences. We computed the cosine similarity based on the tf-idf representation of a pair of sentences. α is set to 2.0 according to the performance on the validation set.

4.2 Quality of the Generated Explanations

To comprehensively evaluate the quality of the generated explanations at both word-level and attribute-level, we employed different types of metrics, including BLEU-{1, 2, 4} and F1 score of ROUGE-{1,2,L}, which are used to measure word-level content generation quality, and Precision, Recall and F1 score of mentioned attributes, which are used to measure the attribute-level content generation quality. The results are reported in Table 3 and Table 4 respectively.

Word-level Content Generation Quality. GREENer outperformed baselines under every BLEU and ROUGE metric on both datasets. As BLEU is a precision-based metric, a larger BLEU achieved by a model suggests that a larger portion of content in sentences

[1] https://www.gurobi.com/products/gurobi-optimizer/

Table 3: Comparison of word-level explanation quality by different models on Ratebeer and TripAdvisor.

Model	BLEU(%)			ROUGE(%)		
	1	2	4	1	2	L
Ratebeer						
NRT	27.03	11.97	2.50	27.16	4.83	24.63
SAER	28.40	12.68	2.66	27.59	4.92	25.29
NARRE	24.67	9.09	1.41	22.13	2.79	20.04
SEER	14.24	5.77	1.03	20.18	2.74	21.08
ESCOFILT	26.36	12.27	3.55	27.55	5.58	24.62
GREENer	**36.59**	**19.14**	**6.15**	**33.03**	**8.34**	**29.84**
TripAdvisor						
NRT	20.30	8.31	1.70	20.25	2.92	18.86
SAER	20.94	8.80	1.90	20.67	3.23	19.23
NARRE	22.22	8.59	1.67	21.92	2.85	19.52
SEER	21.84	9.00	1.89	21.77	3.13	20.44
ESCOFILT	22.81	9.40	2.39	22.94	3.41	20.42
GREENer	**28.78**	**13.39**	**4.38**	**24.99**	**4.90**	**22.33**

Table 4: Comparison of attribute-level explanation quality by different models on Ratebeer and TripAdvisor.

Model	Attribute Prediction(%)					
	P	R	F1	P	R	F1
Dataset	Ratebeer			TripAdvisor		
NRT	29.27	23.72	24.76	22.82	19.22	17.82
SAER	28.82	23.75	24.57	**22.95**	19.23	18.02
NARRE	20.19	24.38	20.08	17.32	24.95	18.21
SEER	**31.45**	22.57	24.17	22.66	26.15	21.95
ESCOFILT	21.32	32.99	24.39	16.92	28.21	19.26
GREENer	30.60	**40.02**	**32.74**	20.92	**33.68**	**23.46**

selected by the model is relevant to the ground-truth sentences. In other words, more content in sentences selected by GREENer can reflect/predict the users' opinions towards the recommended items. RNN-based generative methods such as NRT and NARRE suffer from generating short and generic content, such as "*the staff was very friendly and helpful.*" Although such generic content has a larger chance to match with the ground-truth sentences, the brevity penalty in BLEU penalizes methods focusing too much on such less information sentences. The comparison of GREENer against NARRE shows that though both methods extract sentences from an existing corpus, the graph structure employed in GREENer can better recognize sentences matching users' criteria of items than the attention structure used in NARRE. Both GREENer and SEER select the final K sentences using ILP. The comparison between them shows that GREENer's learnt sentence relevance score better reflects the utility of a sentence as an explanation than directly modeling it from pre-computed user sentiment scores over item aspects. ESCOFILT is enforced to cover K different aspects of an item for its explanations to all users, while GREENer learns to predict what the most important sentences are for each user-item pair. This hard requirement in ESCOFILT makes it one of the most competitive baselines, but its lack of flexibility also leaves it behind GREENer. It is noteworthy that GREENer achieves much larger BLEU-4 than all baselines. It suggests the explanations generated by GREENer not only have more overlaps with the ground-truth, but also cover longer segments in the ground-truth; and therefore its content is more coherent as a whole.

On the other hand, as ROUGE is a recall-based metric, a higher ROUGE score achieved by a model suggests that more content in ground-truth sentences are included in the sentences selected by the model. This further demonstrates the effectiveness of GREENer in recognizing the relevance of candidate sentences to a user-item pair. In particular, GREENer achieved much higher ROUGE-L than all baselines. This indicates that GREENer is more successful in identifying sentences with long consecutive spans that match users' comments about items. The graph structure that fuses heterogeneous information from users, items, attributes and sentences contributes the most to GREENer's better extraction performance, which will be validated in our ablation analysis in Section 4.3.

Attribute-level Content Generation Quality. To better understand whether the explanations generated by GREENer cover more important information about the target items' attributes, we also evaluated the Precision, Recall and F1 score of the attributes contained in the synthesized explanation with respect to the corresponding ground-truth. As we can observe in Table 4, GREENer outperformed baselines with a large margin in recall and comparable precision on both datasets. This strongly suggests GREENer can better recognize the relevance of candidate sentences with respect to the important item attributes, and maximize the coverage of those attributes. On the other hand, it confirms that GREENer's encouraging performance on BLEU and ROUGE is not simply because of its extractive nature, but also due to its ability to more accurately identify and cover important target attributes.

4.3 Ablation Analysis

We include four variants of our solution to study the contribution of each component to the performance of GREENer:

- ¬ *GAT.* This variant excludes the graph neural network model and only preserves the user, item, attribute and sentence information. For each sentence s^i from $\mathcal{S}_u \bigcup \mathcal{S}_c$, we concatenate its embedding with user u, item c and mean-pooling of f_{s^i}, where f_{s^i} represents the item attributes in sentence s^i. Then the concatenated embedding is directly fed into DCN to predict the relevance score of sentence s^i. The comparison between this variant and GREENer indicates the necessity of modeling the user, item, attributes and sentences into a heterogeneous graph to learn their intermediate relationships.
- ¬ *BERT.* This variant replaces BERT with the average word embeddings for sentence representations. The comparison between this variant and GREENer demonstrates the importance of using a pre-trained language model to encode sentences as input sentence node representations.
- ¬ *DCN.* This variant replaces DCN in GREENer with a single linear layer. The comparison between this variant and GREENer presents the effectiveness of including direct feature-level interaction among user, item and sentences when learning the final sentence representations.
- ¬ *ILP.* This variant replaces the ILP-based sentence selection strategy with a vanilla strategy which selects sentences solely based on the predicted probabilities of sentences in a descending order. The comparison between this variant and GREENer shows

Table 5: Example explanations produced by different models on Ratebeer and TripAdvisor.

Model	Explanation
Ratebeer	
Ground-Truth	Roasty, chocolate malts paired with chinook hops. Very smooth sipper with a nice balance between sweet, smooth malt, and tangy hops.
NARRE	Medium carbonation and body; with a very nice creamy and slightly slick mouthfeel.
SAER	A dark orange color with a medium - sized white head.
GREENer	Tastes of **hops** and **roasty chocolate**.Taste is very **smooth** with bitter and herbal **hops** with creamy toasted **malts**.
TripAdvisor	
Ground-Truth	The rooms are the perfect size and have everything you would need. The outdoor hot tub is also fantastic for your aching legs after a full day of skiing.
NARRE	The hotel is very centrally located so we were able to walk which was really nice.
SAER	Great hotel for breakfast, and the staff are very friendly. **Room** is clean and spacious.
GREENer	The **hot tub** is great and a good place to meet people. Comfortable beds, nice large bathrooms. maybe the cleanest motel **room** I've ever been in.

the utility of considering both sentence relevance and redundancy when selecting sentences as explanations.

Table 6: Ablation analysis on Ratebeer dataset.

Model	BLEU (%)			ROUGE (%)		
	1	2	4	1	2	L
Ratebeer						
GREENer	**36.59**	**19.14**	**6.15**	**33.03**	**8.34**	**29.84**
¬ GAT	33.95	17.74	5.88	31.25	8.14	28.23
¬ BERT	33.26	16.59	4.96	32.16	7.40	28.88
¬ DCN	35.13	17.63	5.35	31.87	7.31	29.08
¬ ILP	27.73	14.79	5.75	24.59	7.46	26.01

The results of our ablation analysis about GREENer are reported in Tabel 6. The same as Table 3, we report BLEU-{1, 2, 4} and F1 score of ROUGE-{1,2,L}. All the variants performed worse than GREENer. The most important component turns out to be ILP, which is expected. As the learnt sentence embedding in GREENer reflects the relevance of a sentence to a given user-item pair independently from other sentences, simply counting on this sentence representation for selecting sentences can hardly avoid repetitions. When we looked into the output of ¬ ILP, most of its top ranked sentences are very similar to each other and therefore can hardly cover a comprehensive set of aspects of the target item. The next most important component is the heterogeneous graph structure, which helps GREENer capture the complex relationship between a user's preference and item's aspects, which cannot be tackled by directly performing feature crossing. In addition, using BERT to obtain sentence representations enables GREENer to learn a better sentence representation. This analysis suggested that DCN introduced least impact on GREENer's final performance; but combining DCN with GAT can still boost the model's performance. This indicates feature-level interactions in the embedding space provide a complementary view for the final sentence representation and selection.

4.4 Case Study

We present two group of example explanations produced by GREENer and other baselines in Table 5. The ground-truth explanations are also included for reference. We manually labeled the overlapping attributes in all the generated sentences (w.r.t the ground-truth). From the table, we can clearly observe that the extracted sentences by GREENer are much more relevant to the ground-truth explanations. The attributes in extracted sentences match those in ground-truth explanations, especially those uncommon attributes such as "roasty chocolate" in the first example and "hot tub" in the second example. With the explicitly mentioning of attributes, the explanations generated by GREENer can help users make more informed decisions on which item better suits their preferences and needs.

5 CONCLUSION AND FUTURE WORK

In this paper, we present a graph neural network based extractive solution for explaining a system's recommended items to its users. It integrates heterogeneous information about user, item, item attributes and candidate sentences to evaluate the relevance of a sentence with respect to a particular user-item pair. Item attributes are introduced as the intermediary to address sparsity in observations at the user-item level, and for the same purpose pre-trained language models are used to encode item attributes and sentences for the model learning. Finally, to optimize the trade-off among individual sentence relevance, overall attribute coverage and content redundancy, we solve an integer linear programming problem to make the final selection of sentences for a user-item pair. Extensive experiment comparisons against a set of state-of-the-art explanation methods demonstrate the advantages of our solution in providing high-quality explanation content.

We should note extraction-based explanation methods still have their intrinsic limitations: their input is restricted to an item's existing reviews; for items with limited exposure, e.g., a new item, such solutions (including ours) cannot provide any informative explanations. Our current attempt to address this limitation was to incorporate the same user's historical reviews about items from the same category. Leveraging generative solutions to synthesize explanations could be a potential choice in such situations. In addition, currently the scoring function's weights in our ILP were manually set. Learning-based methods can be introduced to optimize it for better performance in our future work.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under grant IIS-1553568, IIS-1718216 and IIS-2007492.

REFERENCES

[1] Charu C Aggarwal et al. 2016. *Recommender systems*. Vol. 1. Springer.

[2] Mustafa Bilgic and Raymond J Mooney. 2005. Explaining recommendations: Satisfaction vs. promotion. In *Beyond Personalization Workshop, IUI*, Vol. 5.

[3] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).

[4] Renqin Cai, Xueying Bai, Zhenrui Wang, Yuling Shi, Parikshit Sondhi, and Hongning Wang. 2018. Modeling sequential online interactive behaviors with temporal point process. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 873–882.

[5] Renqin Cai, Chi Wang, and Hongning Wang. 2017. Accounting for the Correspondence in Commented Data. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 365–374.

[6] Renqin Cai, Qinglei Wang, Chong Wang, and Xiaobing Liu. 2020. Learning to structure long-term dependence for sequential recommendation. *arXiv preprint arXiv:2001.11369* (2020).

[7] Renqin Cai, Jibang Wu, Aidan San, Chong Wang, and Hongning Wang. 2021. Category-aware collaborative sequential recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 388–397.

[8] 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*. 1583–1592.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[10] Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2020. Evaluating the state-of-the-art of end-to-end natural language generation: The e2e nlg challenge. *Computer Speech & Language* 59 (2020), 123–156.

[11] Daniel Fleder and Kartik Hosanagar. 2009. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management science* 55, 5 (2009), 697–712.

[12] Jonathan L Herlocker, Joseph A Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. ACM, 241–250.

[13] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[14] Trung-Hoang Le and Hady W Lauw. 2021. Synthesizing aspect-driven recommendation explanations from reviews. IJCAI.

[15] Chenliang Li, Cong Quan, Li Peng, Yunwei Qi, Yuming Deng, and Libing Wu. 2019. A Capsule Network for Recommendation and Explaining What You Like and Dislike. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 275–284.

[16] Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 755–764.

[17] Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017. Neural rating regression with abstractive tips generation for recommendation. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 345–354.

[18] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[19] Weizhi Ma, Min Zhang, Yue Cao, Woojeong Jin, Chenyang Wang, Yiqun Liu, Shaoping Ma, and Xiang Ren. 2019. Jointly learning explainable rules for recommendation with knowledge graph. In *The World Wide Web Conference*. 1210–1221.

[20] Julian McAuley, Jure Leskovec, and Dan Jurafsky. 2012. Learning attitudes and attributes from multi-aspect reviews. In *2012 IEEE 12th International Conference on Data Mining*. IEEE, 1020–1025.

[21] Silvia Milano, Mariarosaria Taddeo, and Luciano Floridi. 2020. Recommender systems and their ethical challenges. *AI & SOCIETY* 35, 4 (2020), 957–967.

[22] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 188–197.

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 311–318.

[24] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[25] Reinald Adrian Pugoy and Hung-Yu Kao. 2021. Unsupervised Extractive Summarization-Based Representations for Accurate and Explainable Collaborative Filtering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2981–2990.

[26] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. (2018).

[27] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1135–1144.

[29] Amit Sharma and Dan Cosley. 2013. Do social explanations work?: studying and modeling the effects of social explanations in recommender systems. In *Proceedings of the 22nd international conference on World Wide Web*. 1133–1144.

[30] Rashmi Sinha and Kirsten Swearingen. 2002. The role of transparency in recommender systems. In *CHI'02 extended abstracts on Human factors in computing systems*. ACM, 830–831.

[31] Kirsten Swearingen and Rashmi Sinha. 2001. Beyond algorithms: An HCI perspective on recommender systems. In *ACM SIGIR 2001 Workshop on Recommender Systems*, Vol. 13. Citeseer, 1–11.

[32] Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. 2008. Providing justifications in recommender systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 38, 6 (2008), 1262–1272.

[33] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. *arXiv preprint arXiv:2108.10539* (2021).

[34] Yiyi Tao, Yiling Jia, Nan Wang, and Hongning Wang. 2019. The FacT: Taming Latent Factor Models for Explainability with Factorization Trees. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, New York, NY, USA, 295–304.

[35] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[36] Hongning Wang, Yue Lu, and Chengxiang Zhai. 2010. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 783–792.

[37] Nan Wang, Hongning Wang, Yiling Jia, and Yue Yin. 2018. Explainable recommendation via multi-task learning in opinionated text data. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 165–174.

[38] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.

[39] Xiting Wang, Yiru Chen, Jie Yang, Le Wu, Zhengtao Wu, and Xing Xie. 2018. A reinforcement learning framework for explainable recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 587–596.

[40] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2019. Explainable reasoning over knowledge graphs for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5329–5336.

[41] Jibang Wu, Renqin Cai, and Hongning Wang. 2020. Déjà vu: A contextualized temporal attention mechanism for sequential recommendation. In *Proceedings of The Web Conference 2020*. 2199–2209.

[42] Yikun Xian, Zuohui Fu, Shan Muthukrishnan, Gerard De Melo, and Yongfeng Zhang. 2019. Reinforcement knowledge graph reasoning for explainable recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. 285–294.

[43] Yikun Xian, Tong Zhao, Jin Li, Jim Chan, Andrey Kan, Jun Ma, Xin Luna Dong, Christos Faloutsos, George Karypis, Shan Muthukrishnan, et al. 2021. EX3: Explainable Attribute-aware Item-set Recommendations. In *Fifteenth ACM Conference on Recommender Systems*. 484–494.

[44] Aobo Yang, Nan Wang, Renqin Cai, Hongbo Deng, and Hongning Wang. 2021. Comparative Explanations of Recommendations. *arXiv preprint arXiv:2111.00670* (2021).

[45] Aobo Yang, Nan Wang, Hongbo Deng, and Hongning Wang. 2021. Explanation as a Defense of Recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 1029–1037.

[46] Yongfeng Zhang and Xu Chen. 2018. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192* (2018).

[47] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 83–92.

[48] Yongfeng Zhang, Haochen Zhang, Min Zhang, Yiqun Liu, and Shaoping Ma. 2014. Do users rate or review? Boost phrase-level sentiment labeling with review-level sentiment classification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 1027–1030.