Unsupervised Extractive Summarization-Based Representations for Accurate and Explainable Collaborative Filtering

Reinald Adrian Pugoy[1,2] and **Hung-Yu Kao**[1]
[1]Intelligent Knowledge Management Lab
National Cheng Kung University, Tainan City, Taiwan
[2]Faculty of Information and Communication Studies
University of the Philippines Open University, Los Baños, Philippines
rdpugoy@up.edu.ph, hykao@mail.ncku.edu.tw

Abstract

We pioneer the first extractive summarization-based collaborative filtering model called ESCOFILT. Our proposed model specifically produces extractive summaries for each item and user. Unlike other types of explanations, summary-level explanations closely resemble real-life explanations. The strength of ESCOFILT lies in the fact that it unifies representation and explanation. In other words, extractive summaries both represent and explain the items and users. Our model uniquely integrates BERT, K-Means embedding clustering, and multilayer perceptron to learn sentence embeddings, representation-explanations, and user-item interactions, respectively. We argue that our approach enhances both rating prediction accuracy and user/item explainability. Our experiments illustrate that ESCOFILT's prediction accuracy is better than the other state-of-the-art recommender models. Furthermore, we propose a comprehensive set of criteria that assesses the real-life explainability of explanations. Our explainability study demonstrates the superiority of and preference for summary-level explanations over other explanation types.

1 Introduction

Collaborative filtering (CF) approaches are the most dominant and outstanding models in recommender systems literature. CF mainly focuses on learning accurate representations of users and items, denoting user preferences and item characteristics, respectively (Chen et al., 2018; Tay et al., 2018). The earliest CF models learned such representations based on user-given numeric ratings, but employing them is an oversimplification of user preferences and item characteristics (Koren et al., 2009; Musto et al., 2017). In this regard, review texts have been utilized to alleviate this issue.

Reviews Received by the 'Journaling Bible' Item

1. I was not expecting this Bible to be so beautiful when I pre-ordered it 5 months ago, but it arrived in the mail today and it is just gorgeous! I love the concept of Bible journaling, but was always a bit intimidated by where/how to start. This removes that concern through some beautifully done artwork and lettering. I am ecstatic at the quality of this Bible!

2. I brought this as I wanted a separate Bible to do Bible journaling. It is very beautiful and has many images that can be coloured. The pages are similar to Bible paper and cream in colour. Overall a wonderful Bible to do journaling and meditate God's Word.

Generated Explanations

- **Review-Level:** I brought this as I wanted a separate Bible to do Bible journaling. It is very beautiful and has many images that can be coloured. The pages are similar to Bible paper and cream in colour. Overall a wonderful Bible to do journaling and meditate God's Word.

- **Word-Level:** I brought this as I wanted a separate Bible to do Bible journaling. It is very beautiful and has many images that can be coloured. The pages are similar to Bible paper and cream in colour. Overall a wonderful Bible to do journaling and meditate God's Word.

- **Summary-Level:** I was not expecting this Bible to be so beautiful when I pre-ordered it 5 months ago, but it arrived in the mail today and it is just gorgeous! This removes that concern through some beautifully done artwork and lettering. The pages are similar to Bible paper and cream in colour. Overall a wonderful Bible to do journaling and meditate God's Word.

Table 1: Illustration of the different types of explanations. A review-level explanation is simply the highest weighted review. A word-level explanation is comprised of highlighted words or tokens with the highest attention scores. Our proposed summary-level explanation closely resembles real-life explanations, wherein the explanation text is derived from multiple reviews.

The primary benefit of using reviews as the source of features is that they can cover the inherently multi-faceted nature of user opinions. Users can explain their rationales for the ratings they give to items. Thus, reviews contain a large quantity of rich latent information that cannot be otherwise acquired solely from ratings (Chen et al., 2018).

Still, a typical limitation exists for most review-based recommender systems recently; the intrin-

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 2981–2990
August 1–6, 2021. ©2021 Association for Computational Linguistics

sic black-box nature of neural networks (NN) makes the explainability behind predictions obscure (Ribeiro et al., 2016; Wang et al., 2018b). The intricate architecture of hidden layers has opaqued the decision-making processes of neural models (Peake and Wang, 2018). Providing explanations is essential as they could help persuade users to develop further trust in a recommender system and make eventual purchasing decisions (Peake and Wang, 2018; Ribeiro et al., 2016; Zhang et al., 2014).

In light of this, current research efforts have attempted to improve the explainability aspect of recommender systems. Common types of explanations include review-level and word-level. In a review-level explanation, the attention mechanism is applied to measure every review's contribution to the item (or user) embedding (Chen et al., 2018; Feng and Zeng, 2019). High-scoring reviews are then selected to serve as explanations. On the other hand, in a word-level or token-level explanation, informative words in a local window or textual block are selected together (Liu et al., 2019a; Pugoy and Kao, 2020; Seo et al., 2017). Similar to the first mechanism, top words are chosen due to their high attention weights.

Evidently, review-level and word-level explanations are side-effects of applying the attention mechanism to reviews and words. These have been integral and beneficial in formulating better user and item representations. However, we contend that both types of explanations may not completely resemble real-life explanations. In logic, an explanation is a set of intelligible statements usually constructed to describe and clarify the causes, context, and consequences of objects, events, or phenomena under examination (Drake, 2018). Based on our example in Table 1, the review-level explanation is exactly the same as the second item review, assuming that it has the higher attention weight. Due to this, it also inadvertently disregards other possibly useful sentences from other reviews with lower attention scores. Furthermore, even though the word-level explanation contains informative words, it may not be practical in an actual recommendation scenario since it typically appears as fragments. Word-level explanations may not be intelligible enough due to humans' natural bias toward sentences, which are defined to express complete thoughts (Andersen, 2014).

Therefore, in this paper, we propose the first **e**xtractive **s**ummarization-based **co**llaborative **filt**ering model, ESCOFILT. For every item and user, our novel model generates extractive summaries that bear more resemblance to real-life explanations, as seen in Table 1's last row. Unlike a review-level explanation, a summary-level explanation (which we also call *extractive summary*, *representative summary*, and *representation-explanation* in different sections of this paper) is composed of informative statements gathered from different reviews. As opposed to a word-level explanation, an ESCOFILT-produced explanation is more comprehensible as it can convey complete thoughts. It should be noted that our model performs extractive summarization in an unsupervised manner since expecting ground-truth summaries for all items and users in a large dataset is unrealistic. The strength of ESCOFILT lies in the fact that it uniquely unifies representation and explanation. In other words, an extractive summary both *represents* and *explains* a particular item (or user). We argue that our approach enhances both rating prediction accuracy and user/item explainability, which are later validated by our experiments and explainability study.

1.1 Contributions

These are the main contributions of our paper:

- To the best of our knowledge, we pioneer the first extractive summarization-based CF framework.
- Our proposed model uniquely integrates BERT, K-Means embedding clustering, and multilayer perceptron (MLP) to respectively learn sentence embeddings, extractive representation-explanations, and user-item interactions.
- To the extent of our knowledge, ESCOFILT is one of the first recommender models that employ BERT as a review feature extractor.
- We also propose a comprehensive set of criteria that assesses the explainability of explanation texts in real life.
- Our experiments illustrate that the rating prediction accuracy of ESCOFILT is better than the other state-of-the-art models. Moreover, our explainability study shows that summary-level explanations are superior and more preferred than the other types of explanations.

2 Related Work

Developing a CF model involves two crucial steps, i.e., learning user and item representations and modeling user-item interactions based on those representations (He et al., 2018). One of the foundational works in utilizing NN for CF is neural collaborative filtering or NCF (He et al., 2017). Originally implemented for implicit feedback data-driven CF, NCF learns non-linear interactions between users and items by employing MLP layers as its interaction function.

DeepCoNN is the first deep learning-based model representing users and items from reviews in a coordinated manner (Zheng et al., 2017). The model consists of two parallel networks powered by convolutional neural networks (CNN). One network learns user behavior by examining all reviews he has written, and the other network models item properties by exploring all reviews it has received. A shared layer connects these two networks, and factorization machines capture user-item interactions. Another notable model is NARRE, which shares several similarities with DeepCoNN. NARRE is also composed of two parallel CNN-based networks for user and item modeling (Chen et al., 2018). For the first time, this model incorporates the review-level attention mechanism that determines each review's usefulness or contribution based on attention weights. As a side-effect, this also leads to review-level explanations; reviews with the highest attention scores are presented as explanations. These weights are then integrated into the representations of users and items to enhance embedding quality and prediction accuracy.

Other related studies include D-Attn (Seo et al., 2017), MPCN (Tay et al., 2018) DAML (Liu et al., 2019a), and HUITA (Wu et al., 2019). These all employ different types of attention mechanisms to distinguish informative parts of a given data sample, resulting in simultaneous accuracy and explainability improvements. D-Attn integrates global and local attention to score each word to determine its relevance in a review text. MPCN is similar to NARRE, but the former relies solely on attention mechanisms without any need for convolutional layers. DAML utilizes CNN's local and mutual attention to learn review features, and HUITA incorporates a hierarchical, three-tier attention network.

Most of these aforementioned models take advantage of CNNs as automatic review feature extractors. Coupling them with mainstream word embeddings leads to the formulation of user and item representations. However, such approaches fail to consider global context and word frequency information. The two said factors are crucial as they can affect recommendation performance (Pilehvar and Camacho-Collados, 2019; Wang et al., 2018a). To deal with such dilemmas, NCEM (Feng and Zeng, 2019) and BENEFICT (Pugoy and Kao, 2020) use a pre-trained BERT model to obtain review features. BERT's advantage lies in its full retention of global context and word frequency information (Feng and Zeng, 2019). For explainability, NCEM similarly adopts NARRE's review-level attention. On the contrary, BENEFICT utilizes BERT's self-attention weights in conjunction with a solution to the maximum subarray problem (MSP). BENEFICT's approach produces an explanation based on a subarray of contiguous tokens with the largest possible sum of self-attention weights.

In summary, there appears to be a trend; tackling explainability improves prediction and recommendation performance consequentially. While most recommender models address this via attention mechanisms, our proposed model solves this by unifying representation and explanation in the form of extractive summaries. As evidenced in the succeeding sections of this paper, we argue that our approach can further enhance CF's accuracy and explainability.

3 Methodology

ESCOFILT, whose architecture is illustrated in Figure 1, has two parallel components that learn summarization-based user and item representations. From Sections 3.2 to 3.3, we will only discuss the item modeling process as it is nearly identical to user modeling, with their inputs as the only difference.

3.1 Definition and Notation

The training dataset τ consists of N tuples, with the latter denoting the size of the dataset. Each tuple follows this form: (u, i, r_{ui}, v_{ui}) where r_{ui} and v_{ui} respectively refer to the ground-truth rating and review accorded by user u to item i. Moreover, let $V_u = \{v_{u1}, v_{u2}, ..., v_{uj}\}$ be the set of all j reviews written by user u. Similarly, let $V_i = \{v_{1i}, v_{2i}, ..., v_{ki}\}$ be the set of all k reviews received by item i. Both V_u and V_i are obtained from scanning τ itself.

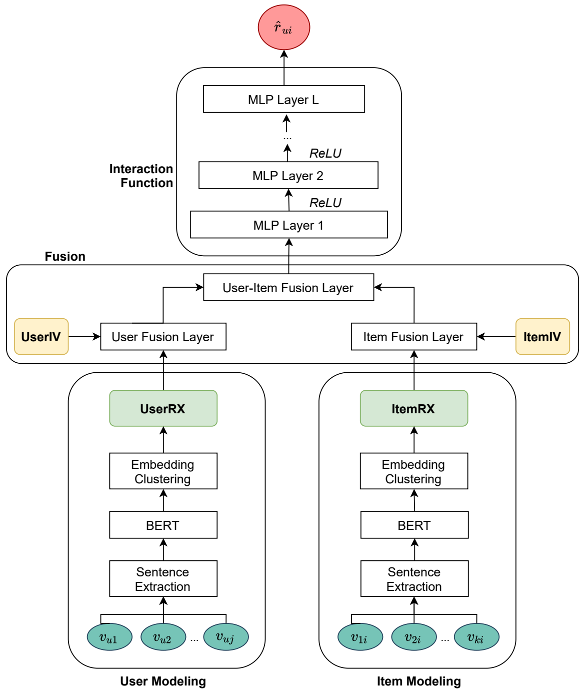

Figure 1: The proposed ESCOFILT architecture.

The input of ESCOFILT is a user-item pair (u, i) from each tuple in τ. We particularly feed V_u and V_i to the model as they initially represent u and i. The output is the predicted rating $\hat{r}_{ui} \in \mathbb{R}$ that user u may give to item i. Thus, the rating prediction task R can be expressed as:

$$R(u, i) = (V_u, V_i) \rightarrow \hat{r}_{ui} \tag{1}$$

Its corresponding objective function, the mean squared error (MSE), is given below:

$$MSE = \frac{1}{|\tau|} \sum_{u,i \in \tau} (r_{ui} - \hat{r}_{ui})^2 \tag{2}$$

3.2 Sentence Extraction and BERT Encoding

First, the reviews in V_i are concatenated together to form a single document. A sentence segmentation component called Sentencizer (by spaCy) is utilized to split this document into individual sentences (Gupta and Nishu, 2020). The set of all sentences in V_i is now given by $S_i = \{s_{i1}, s_{i2}, ..., s_{ig}\}$ where g refers to the total number of sentences.

Afterward, S_i is fed to a pre-trained BERT$_{\text{LARGE}}$ model. It should be noted that we opt not to use `[CLS]` representations as these may not necessarily provide the best sentence embeddings (Miller, 2019). In this regard, we tap BERT's penultimate encoder layer to obtain the contextualized word embeddings. The word embeddings of each sentence in S_i are stored in $\bar{S}_i \in \mathbb{R}^{g \times w \times 1024}$; w pertains to the amount of words in a sentence, and 1024 is the embedding size of BERT. Then, we average every sentence's word embeddings in $\bar{S}_i$ to produce the set of sentence embeddings $S'_i = \{s'_{i1}, s'_{i2}, ..., s'_{ig}\}$, with $S'_i \in \mathbb{R}^{g \times 1024}$.

3.3 Embedding Clustering

K-Means clustering is next performed to partition the sentence embeddings in $S'i$ into K clusters. Its objective is to minimize the intra-cluster sum of the distances from each sentence to its nearest centroid, given by the following equation (Xia et al., 2020):

$$J_i = \sum_{x=1}^{K} \sum_{s'_{iy} \in C_x} ||s'_{iy} - c_x||^2 \tag{3}$$

where c_x is the centroid of cluster C_x that is closest to the sentence embedding s'_{iy}. The objective function J_i is optimized for item i by running the assignment and update steps until the cluster centroids stabilize. The assignment step assigns each sentence to a cluster based on the shortest sentence embedding-cluster centroid distance, provided by the formula below:

$$d(s'_{iy}) = argmin_{x=1,...,K}\{||s'_{iy} - c_x||^2\} \tag{4}$$

where d is a function that obtains the cluster closest to s'_{iy}. Furthermore, the update step recomputes the cluster centroids based on new assignments from the previous step. This is defined as:

$$c_x = \frac{1}{|C_x|} \sum_{y=1}^{g} \{s'_{iy} | d(s'_{iy}) = x\} \tag{5}$$

where $|C_x|$ refers to the number of sentences that cluster C_x contains. By introducing clustering, redundant and related sentences are grouped in the same cluster. Concerning this, K is derived using this equation:

$$K = \phi_i \times g \tag{6}$$

where ϕ_i pertains to the item summary ratio, i.e., the percentage of sentences that comprise an item's extractive summary. This subsequently implies that K denotes the actual number of sentences in the summary. Sentences closest to each cluster centroid are selected and combined to form the item's representation-explanation. This is mathematically expressed as:

$$e(C_x) = argmin_{y=1,...,g}\{||s'_{iy} - c_x||^2\}$$
$$ItemRX_i = \frac{1}{K}\sum_{x=1}^{K} s'_{i,e(C_x)} \tag{7}$$

where e is a function that returns the nearest sentence to the centroid c_x of cluster C_x, and $ItemRX_i \in \mathbb{R}^{1\times 1024}$ is the representation-explanation embedding of item i.

3.4 Fusion Layers

Inspired by NARRE (Chen et al., 2018), we also draw some principles from the traditional latent factor model by incorporating rating-based hidden vectors that depict users and items to a certain extent. These are represented by $UserIV$ and $ItemIV$, both in $\mathbb{R}^{1\times m}$ where m is the dimension of the latent vectors. Such vectors are fused with their respective representation-explanation embeddings. This is facilitated by these fusion levels, illustrated by the following formulas:

$$\begin{aligned} f_u &= (UserRX_u \times W_u + b_u) + UserIV_u \\ f_i &= (ItemRX_i \times W_i + b_i) + ItemIV_i \\ f_{ui} &= [f_u, f_i] \end{aligned} \tag{8}$$

where f_u and f_i pertain to the preliminary fusion layers and both are in $\mathbb{R}^{1\times m}$; W_u and W_i are weight matrices in $\mathbb{R}^{1024\times m}$; b_u and b_i refer to bias vectors; and $f_{ui} \in \mathbb{R}^{1\times 2m}$ denotes the initial user-item interactions from the third fusion layer and is later fed to the MLP.

3.5 Multilayer Perceptron and Rating Prediction

The MLP is necessary to model the CF effect, i.e., to learn meaningful non-linear interactions between users and items. An MLP with multiple hidden layers typically implies a higher degree of non-linearity and flexibility. Similar to the strategy of He et al. (2017), ESCOFILT adopts an MLP with a tower pattern; the bottom layer is the widest while every succeeding top layer has fewer neurons. A tower structure enables the MLP to learn more abstractive data features. Specifically, we halve the size of hidden units for each successive higher layer. ESCOFILT's MLP component is defined as follows:

$$\begin{aligned} h_1 &= ReLU(f_{ui} \times W_1 + b_1) \\ h_L &= ReLU(h_{L-1} \times W_L + b_L) \end{aligned} \tag{9}$$

Dataset	#Reviews	#Users	#Items
Automotive	20,473	2,928	1,835
Digital Music	64,706	5,541	3,568
Instant Video	37,126	5,130	1,685
Patio, Lawn, & Garden	13,272	1,686	962

Table 2: Statistics of the datasets utilized in our study.

where h_L represents the L-th MLP layer, and W_L and b_L pertain to the L-th layer's weight matrix and bias vector, respectively. As far as the MLP's activation function is concerned, we select the rectified linear unit (ReLU), which yields better performance than other activation functions (He et al., 2017). Finally, the MLP's output is fed to one more linear layer to produce the predicted rating:

$$\hat{r}_{ui} = h_L \times W_{L+1} + b_{L+1} \tag{10}$$

4 Empirical Evaluation

4.1 Research Questions

In this section, we detail our experimental setup designed to answer the following research questions (RQs):

- **RQ1:** Does ESCOFILT outperform the other state-of-the-art recommender baselines?
- **RQ2:** Is embedding clustering effective?
- **RQ3:** Can our model produce explanations acceptable to humans in real life?

4.2 Datasets, Baselines, and Evaluation Metric

Table 2 summarizes the four public datasets[1] that we utilized in our study. These datasets are Amazon 5-core, wherein users and items are guaranteed to have at least five reviews each (McAuley et al., 2015; He and McAuley, 2016). The ratings across all datasets are in the range of [1, 5]. We split each dataset into training (80%), validation (10%), and test (10%) sets. Next, to validate the effectiveness of ESCOFILT, we compared its prediction performance against four state-of-the-art baselines:

- **BENEFICT** (Pugoy and Kao, 2020): This recent recommender model uniquely integrates BERT, MSP, and MLP to learn representations, explanations, and interactions.

[1] http://jmcauley.ucsd.edu/data/amazon/

- **DeepCoNN** (Zheng et al., 2017): This is the first deep collaborative neural network model that is based on two parallel CNNs to jointly learn user and item features.
- **MPCN** (Tay et al., 2018): Akin to NARRE, this CNN-less model employs a new type of dual attention for identifying relevant reviews.
- **NARRE** (Chen et al., 2018): Similar to DeepCoNN, it is a neural attentional regression model that integrates two parallel CNNs and the review-level attention mechanism.

All these recommender models employed the same dataset split. We then computed the root mean square error (RMSE) on the test dataset (τ), as indicated by the formula below. RMSE is a widely used metric for evaluating a model's rating prediction accuracy (Steck, 2013).

$$RMSE = \sqrt{\frac{1}{|\tau|} \sum_{u,i \in \tau} (r_{ui} - \hat{r}_{ui})^2} \qquad (11)$$

4.3 Experimental Settings

For ESCOFILT, we mainly based its summarization component on BERT Extractive Summarizer[2] by Miller (2019). We also utilized the pre-trained $\text{BERT}_{\text{LARGE}}$ model afforded by the Transformers library of HuggingFace[3]. In our implementation[4], the following hyperparameters were fixed:

- Learning rate: 0.006
- Quantity of MLP layers: 4
- Item summary ratio (ϕ_i): 0.4
- User summary ratio (ϕ_u): 0.4

On the other hand, we operated an exhaustive grid search over these hyperparameters:

- Number of epochs: [1, 30]
- Latent vector dimension (m): {32, 128, 220}

Due to its architectural similarity to ESCOFILT, we reimplemented BENEFICT by augmenting it with the pre-trained $\text{BERT}_{\text{LARGE}}$ model and adopting our model's fusion and latent vector dimension strategies. For DeepCoNN, MPCN, and NARRE, we employed the extensible NRRec framework[5] and retained the other hyperparameters reported in the framework (Liu et al., 2019b).

For the four baselines, we also performed an exhaustive grid search over the following:

- Number of epochs: [1, 30]
- Learning rates: {0.003, 0.004, 0.006}

All models, including ESCOFILT, used the same optimizer, Adam, which leverages the power of adaptive learning rates during training (Kingma and Ba, 2014). This makes the selection of a learning rate less cumbersome, leading to faster convergence (Chen et al., 2018). Without special mention, the models shared the same random seed, batch size (128), and dropout rate (0.5). We selected the model configuration with the lowest RMSE on the validation set. We ran our experiments on NVIDIA GeForce RTX 2080 Ti.

4.4 Prediction Results and Discussion

4.4.1 Performance Comparison

The overall performances of our model and the other baselines are summarized in Table 3. It is essential to remark that although utilizing information derived from reviews is beneficial, a model's performance can vary contingent on how the said information is considered. These are our general findings:

First, our proposed model consistently outperforms all baselines across all datasets. This ascertains the effectiveness of ESCOFILT and clearly answers RQ1. Moreover, this validates our case that coupling BERT (a superior review feature extractor) with embedding clustering enables user and item representations to have finer granularity and fewer redundancies.

Second, receiving the two lowest average RMSE values, BERT-based models (ESCOFILT and BENEFICT) have generally better prediction accuracies than the rest of the mostly CNN-powered baselines. This particular observation verifies the necessity of integrating BERT in a CF architecture. Unlike its mainstream counterparts, BERT produces more semantically meaningful embeddings that keep essential elements such as global context and word frequency information.

4.4.2 Efficacy of Embedding Clustering

This section further discusses the efficacy of K-Means embedding clustering, instrumental in producing user and item representative summaries. Concerning this, we prepared three variants of our model. First is ESCOFILT-N, which does not utilize any embedding clustering. Instead, it relies on

[2] https://github.com/dmmiller612/bert-extractive-summarizer

[3] https://github.com/huggingface/transformers

[4] https://github.com/reinaldncku/ESCOFILT

[5] https://github.com/ShomyLiu/Neu-Review-Rec

Model	Automotive	Digital Music	Instant Video	Patio, Lawn, & Garden	Average
BENEFICT	0.9023	0.8910	0.9746	0.9352	0.9258
DeepCoNN	0.9076	0.8904	0.9778	0.9316	0.9269
MPCN	0.9107	0.9298	0.9976	0.9362	0.9436
NARRE	0.9144	0.8915	0.9758	0.9539	0.9339
ESCOFILT	**0.8968**	**0.8831**	**0.9742**	**0.9298**	**0.9210**

Table 3: Performance comparison of the recommender models. The best RMSE values are boldfaced.

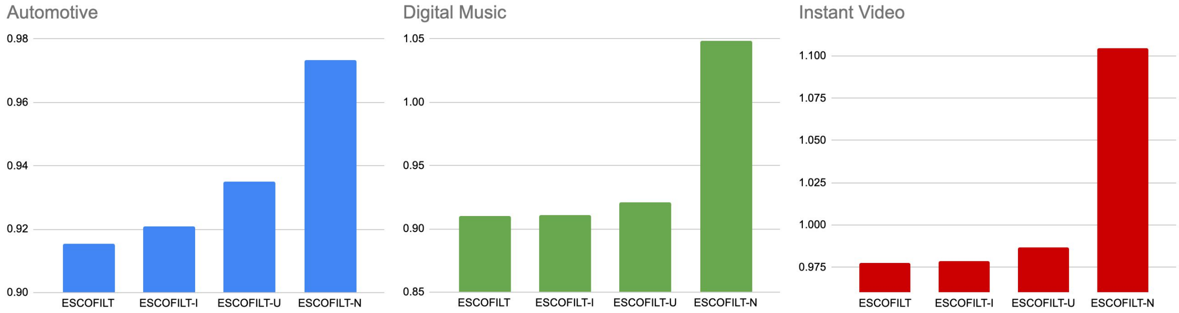

Figure 2: Performance comparison of ESCOFILT variants for illustrating the effectiveness of embedding clustering.

traditional embeddings that are neither pre-trained nor review-based. They are randomly initialized yet optimized during training. Another variant is ESCOFILT-I, wherein only item reviews undergo embedding clustering while the user component is based on traditional embeddings. ESCOFILT-U also operates the same way; the difference is that only user reviews are processed by embedding clustering.

Based on Figure 2, having the lowest validation RMSE values, the default ESCOFILT configuration is the best across the datasets, while the worst variant is ESCOFILT-N. This gives credence to embedding clustering's effectiveness and addresses RQ2; it can simultaneously capture user preferences and item characteristics, resulting in precise representations and accurate rating prediction.

There appears to be a trend as well: the second-best and the third-best variants are ESCOFILT-I and ESCOFILT-U, respectively. In some instances, ESCOFILT-I seems to be on par with the default ESCOFILT variant. This implies that items stand to benefit more than users from embedding clustering. One possible explanation is that each item normally receives a far greater quantity of reviews than each user actually writes, translating to more possibly extractable information and features. Hence, item reviews have a more significant influence than user reviews in determining ratings. Still, this does not immediately suggest that user embedding clustering is not helpful. It needs to be integrated first with item embedding clustering via the MLP to discover relevant user-item interactions, leading to our original model's performance.

5 Explainability Study

5.1 Real-Life Explainability Criteria

The assessment of explanations in existing recommender systems literature is generally limited to specific case studies. Most of these relied on simple qualitative analysis of attention weights and high-scoring reviews on selected samples (Liu et al., 2019a; Seo et al., 2017; Wu et al., 2019). The assessment criterion provided in the NARRE and BENEFICT papers went a little further by asking human raters to score each explanation's helpfulness or usefulness on a given Likert scale (Chen et al., 2018; Pugoy and Kao, 2020). Nevertheless, to the best of our knowledge, there does not appear to be a comprehensive set of criteria that assesses the real-life explainability of explanations. We contend that it is increasingly necessary to measure how people actually perceive explanation texts generated by recommender models; after all, these texts aim to explain entities in real life. Hence, we

Model	Coherence	Completeness	Lack of Alternatives	Novelty	Perceived Truth	Quality	Visualization
BENEFICT	3.52	3.82	3.75	3.58	3.87	3.65	3.65
NARRE	3.68	3.82	**3.82**	3.72	3.75	**3.72**	**3.92**
ESCOFILT	**3.92**	**3.87**	3.73	**3.75**	**3.92**	**3.72**	3.78

Table 4: Comparison of the three explanation types based on the real-life explainability criteria (pointwise evaluation). The best mean values for each criterion are boldfaced.

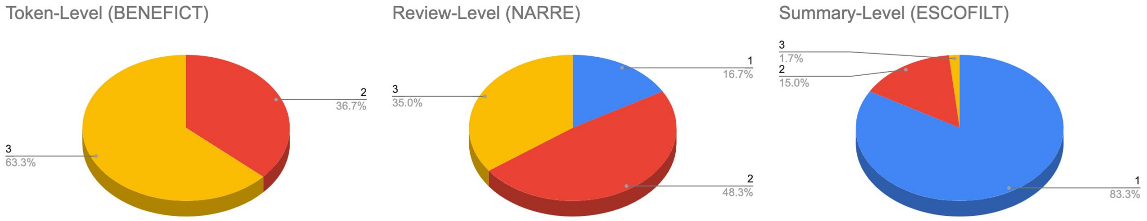

Figure 3: Distribution of the judges' helpfulness rankings for the three explanation types (listwise evaluation).

propose the following explainability criteria, which are inspired by Zemla et al. (2017):

1. **Coherence:** "Parts of the explanation fit together coherently."
2. **Completeness:** "There are no gaps in the explanation."
3. **Lack of Alternatives:** "There are probably less to no reasonable alternative explanations."
4. **Novelty:** "I learned something new from the explanation."
5. **Perceived Truth:** "I believe this explanation to be true."
6. **Quality:** "This is a good explanation."
7. **Visualization:** "It is easy to visualize what the explanation is saying."

5.2 Human Assessment of Explanations

We generated a total of 90 item explanations, 30 each from BENEFICT (token-level), NARRE (review-level), and ESCOFILT (summary-level). For pointwise evaluation, we asked two human judges to assess the explanations based on our proposed real-life explainability criteria on a five-point Likert scale. For listwise evaluation, we instructed them to rank the three explanation types for every text according to helpfulness. We further examined these results by determining the strength of agreement between the two judges, using Cohen's Kappa coefficient (κ) wherein -1 indicates a less than chance agreement, 0 refers to a random agreement, and 1 denotes a perfect agreement (Borromeo and Toyama, 2015; Landis and Koch, 1977).

5.3 Explainability Results and Discussion

Table 4 summarizes the results of the human judges' pointwise evaluation. For five out of seven criteria, ESCOFILT-derived explanations have the highest explainability scores. Specifically, summary-level explanations are most coherent, most complete, most novel, and most truthful. ESCOFILT's strongest aspect is its perceived truth, obtaining a mean rating of 3.92 and $\kappa = 0.28$ that indicates a fair inter-judge agreement.

Interestingly, both ESCOFILT and NARRE have the best quality, with the same mean rating of 3.72. The Kappa coefficient is 0.11, implying that the judges agree with each other to a certain extent. Considering that a review-level explanation is simply the highest weighted review, our model-generated explanations are assessed on par with the former. Furthermore, review-level explanations have the highest explainability scores in two other criteria, i.e., lack of alternatives and visualization. NARRE's strongest aspect is that its explanations are easiest to visualize, having a mean rating of 3.92 and $\kappa = 0.27$ that denotes a fair inter-judge agreement.

Lastly, Figure 3 shows the results of the human judges' listwise evaluation. Our model produces the most helpful explanations; such explanations are ranked first for almost 83% of the items. These are followed far behind by NARRE's explanations, ranked first for nearly 17% of the items. None of BENEFICT's explanations are ranked first. With $\kappa = 0.45$ for ranking consistency, there is a moderate agreement between the judges.

In summary, these results clearly illustrate the superiority of summary-level explanations in real life that can present necessary guidance to users in making future purchasing decisions, thereby satisfying RQ3.

6 Conclusion and Future Work

In this study, unifying representations and explanations, in the form of extractive summaries, have further enhanced collaborative filtering accuracy and explainability. We have successfully developed a model that uniquely integrates BERT, embedding clustering, and MLP. Our experiments on various datasets verify ESCOFILT's predictive capability, and the human judges' assessments validate its explainability in real life. In the future, we shall consider expanding our model's explainability capability by possibly incorporating other NLP principles such as abstractive summarization and natural language generation.

Acknowledgment

This work was funded in part by Qualcomm through a Taiwan University Research Collaboration Project and also in part by the Ministry of Science and Technology, Taiwan, under NCKU B109-K027D and MOST 109-2221-E-006-173 grants, respectively.

References

Sarah Andersen. 2014. Sentence types and functions. *San Jose State University Writing Center*.

Ria Mae Borromeo and Motomichi Toyama. 2015. Automatic vs. crowdsourced sentiment analysis. In *Proceedings of the 19th International Database Engineering & Applications Symposium*, pages 90–95.

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*, pages 1583–1592.

Jess Drake. 2018. *Introduction to Logic*. ED-Tech Press.

Xingjie Feng and Yunze Zeng. 2019. Neural collaborative embedding from reviews for recommendation. *IEEE Access*, 7:103263–103274.

Sarang Gupta and Kumari Nishu. 2020. Mapping local news coverage: Precise location extraction in textual news content using fine-tuned bert based language model. In *Proceedings of the 4th Workshop on Natural Language Processing and Computational Social Science*, pages 155–162.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, pages 507–517.

Xiangnan He, Xiaoyu Du, Xiang Wang, Feng Tian, Jinhui Tang, and Tat-Seng Chua. 2018. Outer product-based neural collaborative filtering. *arXiv preprint arXiv:1808.03912*.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Donghua Liu, Jing Li, Bo Du, Jun Chang, and Rong Gao. 2019a. DAML: Dual attention mutual learning between ratings and reviews for item recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 344–352.

Hongtao Liu, Fangzhao Wu, Wenjun Wang, Xianchen Wang, Pengfei Jiao, Chuhan Wu, and Xing Xie. 2019b. NRPA: Neural recommendation with personalized attention. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1233–1236.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52.

Derek Miller. 2019. Leveraging BERT for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*.

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 11th ACM Conference on Recommender Systems*, pages 321–325.

Georgina Peake and Jun Wang. 2018. Explanation mining: Post hoc interpretability of latent factor models for recommendation systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2060–2069.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273.

Reinald Adrian Pugoy and Hung-Yu Kao. 2020. BERT-based neural collaborative filtering and fixed-length contiguous tokens explanation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 143–153.

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 & Data Mining*, pages 1135–1144.

Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. 2017. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *Proceedings of the 11th ACM Conference on Recommender Systems*, pages 297–305.

Harald Steck. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Proceedings of the 7th ACM Conference on Recommender systems*, pages 213–220.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018. Multi-pointer co-attention networks for recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2309–2318.

Qianqian Wang, Si Li, and Guang Chen. 2018a. Word-driven and context-aware review modeling for recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1859–1862.

Xiang Wang, Xiangnan He, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2018b. TEM: Tree-enhanced embedding model for explainable recommendation. In *Proceedings of the 2018 World Wide Web Conference*, pages 1543–1552.

Chuhan Wu, Fangzhao Wu, Junxin Liu, and Yongfeng Huang. 2019. Hierarchical user and item representation with three-tier attention for recommendation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1818–1826.

Shuyin Xia, Daowan Peng, Deyu Meng, Changqing Zhang, Guoyin Wang, Elisabeth Giem, Wei Wei, and Zizhong Chen. 2020. A fast adaptive k-means with no bounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Jeffrey C Zemla, Steven Sloman, Christos Bechlivanidis, and David A Lagnado. 2017. Evaluating everyday explanations. *Psychonomic Bulletin & Review*, 24(5):1488–1500.

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 and Development in Information Retrieval*, pages 83–92.

Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, pages 425–434.