1-2020

Synthesizing aspect-driven recommendation explanations from reviews

Trung-Hoang LE
Singapore Management University, thle.2017@smu.edu.sg

Hady W. LAUW
Singapore Management University, hadywlauw@smu.edu.sg

Citation

LE, Trung-Hoang and LAUW, Hady W.. Synthesizing aspect-driven recommendation explanations from reviews. (2020). *Proceedings of the 29th International Joint Conference on Artificial Intelligence, IJCAI 2020, Yokohama, Japan, January 7-11*. 2427-2434.
Available at: https://ink.library.smu.edu.sg/sis_research/5954

Synthesizing Aspect-Driven Recommendation Explanations from Reviews

Trung-Hoang Le and **Hady W. Lauw**
Singapore Management University, Singapore
{thle.2017, hadywlauw}@smu.edu.sg

Abstract

Explanations help to make sense of recommendations, increasing the likelihood of adoption. However, existing approaches to explainable recommendations tend to rely on rigid, standardized templates, customized only via fill-in-the-blank aspect sentiments. For more flexible, literate, and varied explanations covering various aspects of interest, we synthesize an explanation by selecting snippets from reviews, while optimizing for representativeness and coherence. To fit target users' aspect preferences, we contextualize the opinions based on a compatible explainable recommendation model. Experiments on datasets of several product categories showcase the efficacies of our method as compared to baselines based on templates, review summarization, selection, and text generation.

1 Introduction

Explainable recommendations are motivated by the need for not only personalized recommendations, but also the accompanying explanations. Many recommender systems are based on matrix factorization, and the learnt latent factors often lack scrutability (users may not comprehend reasons behind certain recommendations). To induce greater interpretability from the latent factors, the crux of many models (e.g., [Zhang *et al.*, 2014; Wang *et al.*, 2018a]) is to contextualize the latent factors along known aspects and to align a recommendation, and correspondingly its explanation, to aspect sentiments.

Problem. An explanation is typically generated *post hoc* to the recommendation model. Our scope is singularly the provision of explanations, presuming that item recommendation is addressed by a separate model. While there could be various forms of explanation (content-based collaborative filtering, rules [Ma *et al.*, 2019], topics [Tan *et al.*, 2016], or social [Ren *et al.*, 2017], etc.), we focus on natural language explanations, i.e., a collection of sentences highlighting product aspects of interest to the target user. Hence, we assume an *aspect demand* is specified as input, listing the number of sentences required for each aspect. Therefore, the goal is to meet this aspect demand with sentences representative of product quality and user preferences.

Existing works in explainable recommendations rely on templated explanation, i.e., substituting words within a prespecified sentence. For instance, EFM [Zhang *et al.*, 2014] has standardized templates for positive and negative opinions, each time substituting only the [*aspect*], e.g.,:

> You might be interested in [*battery life*], on which this product performs well.
>
> You might be interested in [*lens*], on which this product performs poorly.

To increase variation beyond "well", "poorly", MTER [Wang *et al.*, 2018a] further specifies an <*opinion phrase*>, e.g.,:

> Its [*battery life*] is <*long*>.

As exemplified above, templated explanations could be repetitive, robotic, and limited in their expressiveness. They tend to read less naturally than a human-created sentence.

A product review contains sentences that recount a user's experience with the product, which often go some way towards explaining her choices *post-adoption*. Leveraging this explanatory quality, but intending to explain a predicted recommendation *pre-adoption*, we propose to "synthesize" an explanation by taking snippets from various reviews and putting them together in a coherent manner. Fitted to the recommendation, this synthesis benefits from the expressiveness of human-created review sentences, and yet is still flexible enough to produce varied explanations given the wide array of combinatorial selections from rich review corpora. Moreover, since a candidate sentence may bear in-built sentiment potentially incompatible to a user's own, we expand candidate selection to all aspect-relevant sentences by incorporating opinion contextualization for sentiment compatibility.

Contributions. We make several contributions in this work. As our *first contribution*, we propose a framework called *Synthesizing Explanation for Explainable Recommendation* or SEER. Section 3 describes this framework, expressing the objective and constraints in terms of integer linear programming. As the problem proves NP-hard, our *second contribution* is to further describe a heuristic approximation. Section 4 expands on how the synthesized explanation could be contextualized with compatible opinions. As a *third contribution*, in Section 6 we conduct experiments on four product categories to evaluate the efficacy of our synthesis approach, as opposed to comparative baseline approaches based

$\mathcal{U}, \mathcal{P}, \mathcal{A}, \mathcal{O}, \mathcal{T}$	set of all users, items, aspects, opinions, and reviews
$\mathcal{T}_j \in \mathcal{T}$	set of observed text reviews on product p_j
$\mathcal{S}_j \subseteq \mathcal{T}_j$	set of all sentences on product p_j
$t_{ij} \in \mathcal{T}_j$	a review of user u_i on product p_j
$\mathcal{M}$	explainable recommendation model
Z	aspect-level sentiments
$z_{ijk} \in Z$	sentiment of user u_i on item p_j about aspect a_k
$\mathcal{D}$	aspect demand
τ	solution set of selected sentences
$\Gamma_{ss'}$	variable indicates whether sentence s representing s'
γ_s	variable indicates whether sentence s is selected
$\zeta_{i'}$	variable indicates whether a review $t_{i'j}$ is part of τ
$\sigma_{si'}$	observed indicator of whether sentence s is in $t_{i'j}$
π_{sk}	observed indicator of whether sentence s expresses a_k
$s(w)$	sentence s after substituting opinion phrase w

Table 1: Main Notations

on templates, review summarization and selection, as well as text generation.

2 Problem Formulation

Table 1 lists the notations used in this paper. $\mathcal{U}$ and $\mathcal{P}$ are the universal sets of m users and n products respectively. User $u_i \in \mathcal{U}$ may assign to a product $p_j \in \mathcal{P}$ a rating $r_{ij} \in \mathbb{R}_+$ and a text review t_{ij}. Let R be the observed user-item rating matrix, and $\mathcal{T}$ be the set of observed text reviews. Let $\mathcal{A}$ and $\mathcal{O}$ be the universal sets of aspects and opinion phrases. We assume the occurrence of aspect $a \in \mathcal{A}$ and opinion phrase $o \in \mathcal{O}$ can be detected from a review sentence as described in [Zhang *et al.*, 2014].

Compatible Recommendation Models. Our objective is to synthesize an explanation based on the outputs of compatible explainable recommendation models (see Section 5 for examples). An explainable recommendation model $\mathcal{M}$ produces both personalized recommendations and aspect-level sentiments $Z \in \mathbb{R}_+^{m \times n \times v}$ to facilitate their explanations. $z_{ijk} \in Z$ indicates user u_i's sentiment for aspect a_k of p_j.

Problem Statement. Given aspect-level sentiments Z, and a product p_j recommended to user u_i by a model $\mathcal{M}$, we output an explanation in the form of a collection of sentences τ based on the aspect demand $\mathcal{D}$. Let aspect demand $\mathcal{D} \in \mathbb{N}^v$ be a vector, where each element $\mathcal{D}_k$ is a non-negative integer indicating the number of sentences demanded for aspect $a_k \in \mathcal{A}$, and $v = |\mathcal{A}|$. It follows that the sentences should reflect the aspect-level sentiments of the user specified in Z.

Evaluation. A question arises on how to evaluate a recommendation explanation, aside from the goal of meeting the aspect demand. In the literature, recommendation accuracy is measured in terms of how well the prediction approaches the ground truth (held-out rating). An analogous approach would then be to compare an explanation against a ground truth. Intuitively, the review that a user writes for a product *a posteriori* would have been a "perfect" explanation if we were recommending the same product *a priori*. Thus, in the experiments we will compare synthesized explanations in terms of similarity to held-out reviews.

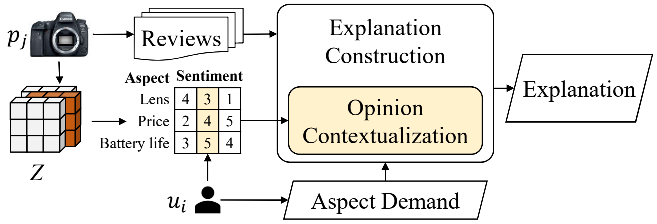

Figure 1: Architecture of proposed framework SEER

3 SEER Framework

As seen in Figure 1, our framework is to synthesize an explanation by selecting snippets (sentences) from a product's existing reviews. Here we discuss the objective of the selection, and offer optimal as well as approximate formulations.

3.1 Optimization Objective

When recommending product p_j to user u_i, we construct an explanation from $\mathcal{T}_j$ (reviews of product p_j). The solution τ ideally consists of $\mathcal{D}_k$ sentences for each $a_k \in \mathcal{A}$, selected from review sentences $\mathcal{S}_j$ (the union of sentences from $\mathcal{T}_j$).

Representativeness. To explain the aspect a_k of p_j well, we aim for the most representative among sentences in $\mathcal{S}_j$ pertinent to a_k. Suppose that how well a sentence s could "represent" another sentence s' is reflected by a cost $\delta_{ss'} \in \mathbb{R}_+$ (lower is better). This may encode application-specific semantic notion of similarity, and for generality we consider these as a given. In Section 6, we experiment with several definitions, including unsupervised (e.g., cosine similarity between *tfidf* vectors), as well as supervised notions (e.g., paraphrase identification, textual entailment [Lan and Xu, 2018]).

Our task is to select $\mathcal{D}_k$ most representative ones to place into the solution set τ. To encode this selection, let $\Gamma_{ss'}$ be a binary variable (the outcome to be determined) indicating whether a selected sentence $s \in \tau$ (i.e., $\gamma_s = 1$) represents another sentence $s' \in \mathcal{S}_j$. We thus want to minimize the representation cost below, where we prefer a solution τ with sentences similar to many of the same aspect.

$$\text{r\_cost}(\tau) = \sum_{s \in \tau} \sum_{s' \in \mathcal{S}_j} \delta_{ss'} \cdot \Gamma_{ss'} \tag{1}$$

Coherence. In addition to capturing the aspects well, the explanation should be compact and coherent. Intuitively, a document by fewer authors would be more coherent than by many. Hence, we attach a cost $\theta_{i'}$ (given) to using a review $t_{i'j} \in \mathcal{T}_j$, rather than to individual sentences. This way, the selection favors selecting sentences that may have come from the same review, presumably enhancing coherence. We define the coherence cost below, where $\zeta_{i'}$ is a binary variable of whether a review $t_{i'j} \in \mathcal{T}_j$ (i.e., $\zeta_{i'} = 1$) is part of the solution set τ (i.e., one or more of its sentences are selected).

$$\text{c\_cost}(\tau) = \sum_{t_{i'j} \in \mathcal{T}_j} \theta_{i'} \cdot \zeta_{i'} \tag{2}$$

The given cost $\theta_{i'}$ also serves to contextualize the explanation to a specific user, as defined shortly in Section 4.

Overall Cost. The overall cost is thus:

$$\text{cost}(\tau) = \text{c\_cost}(\tau) + \text{r\_cost}(\tau) \tag{3}$$

The two components have an inherent trade off. Adding a sentence may lower r_cost if the new sentence is more similar to other sentences, but that risks increasing the c_cost if the new sentence comes from a review not currently in the solution. On the other hand, fewer reviews may constrain the selection of representative sentences. Hence, we need an effective algorithm to find the optimal aggregate of the two.

3.2 Optimal Formulation via ILP

To find an optimal solution τ, we express the problem as Integer Linear Programming (ILP). (4a) is the objective (Eq. 3). γ_s is a binary indicator whether the sentence $s \in \mathcal{S}_j$ is a part of τ. Constraints (4b) and (4c) ensure that sentence $s' \in \mathcal{S}_j$ must be represented by one of the sentences s in the solution set ($\gamma_s = 1$). (4d) means a review must be selected when we select any of its sentences. $\sigma_{si'}$ is an observed binary indicator of whether s is in the review $t_{i'j}$. (4e) ensures a sentence is represented by another of the same aspect. Binary π_{sk} indicates whether s is of aspect a_k. (4f) satisfies aspect demand.

$$\text{min:} \sum_{t_{i'j} \in \mathcal{T}_j} \theta_{i'} \cdot \zeta_{i'} + \sum_{s,s' \in S_j} \delta_{ss'} \cdot \Gamma_{ss'} \tag{4a}$$

$$\text{s.t:} \sum_{s \in \mathcal{S}_j} \Gamma_{ss'} = 1, \forall s' \in S_j \tag{4b}$$

$$\Gamma_{ss'} \leq \gamma_s, \forall s, s' \in \mathcal{S}_j \tag{4c}$$

$$\gamma_s \cdot \sigma_{si'} \leq \zeta_{i'}, \forall t_{i'j} \in \mathcal{T}_j, s \in \mathcal{S}_j \tag{4d}$$

$$\Gamma_{ss'} \leq \sum_{a_k \in \mathcal{A}} \pi_{sk} \cdot \pi_{s'k}, \forall s, s' \in \mathcal{S}_j \tag{4e}$$

$$\sum_{s \in \mathcal{S}_j} \gamma_s \cdot \pi_{sk} = \mathcal{D}_k, \forall a_k \in \mathcal{A} \tag{4f}$$

$$\zeta_{i'}, \gamma_s, \Gamma_{ss'} \in \{0, 1\}, \forall t_{i'j} \in \mathcal{T}_j; \forall s, s' \in \mathcal{S}_j \tag{4g}$$

NP-hardness. Though SEER-ILP is theoretically optimal, it may be intractable for large problem sizes.

Proof. The proof sketch is based on a reduction from the Uncapacitated Facility Location Problem (UFLP) [Cornuéjols *et al.*, 1983] involving a set of facilities and a set of customers. There is a cost to open each facility (favoring fewer facilities) and a cost to serve a customer from an open facility (favoring facility closer to customer). We reduce UFLP to our problem where there is only a single aspect. Each customer is now a sentence s' to be represented. Each facility is a review with opening cost $\theta_{i'}$, associated with one representing sentence s. The service cost is thus $\delta_{ss'}$. Our problem specifies the number of sentences to be selected for that aspect. If we solve for all demands from 1 to m, where m is the total number of facilities, we arrive at a solution for UFLP with the lowest total cost at any number of facilities. Since UFLP is known to be NP-hard, our more general formulation is NP-hard. □

Algorithm 1 SEER-Greedy

```
1: Initialize $\tau = \emptyset$; $S = \mathcal{S}_j$; $T = \mathcal{T}_j$; $D = \mathcal{D}$;
2: while $S \neq \emptyset$ do
3:     for $t_{i'j} \in T$ do
4:         Find $\tau_{i'} \subseteq t_{i'j}$ that represent the most number of unmet aspects in $D$, which minimize the average covering cost of sentences: $\frac{\theta_{i'} + \sum_{s \in \tau_{i'}} \sum_{s' \in S} \delta_{ss'} \cdot \Gamma_{ss'}}{\sum_{s \in \tau_{i'}} \sum_{s' \in S} \Gamma_{ss'}}$
5:     $\tau := \tau \cup \tau_{i'}$; $T := T \backslash t_{i'j}$
6:     $S := S \backslash S'$, where $S'$ are $\tau_{i'}$ covering sentences
7:     $D := D \backslash \{a\}$, where $\{a\}$ are $\tau_{i'}$ representing aspects
8: return $\tau$
```

3.3 Approximation via Greedy Algorithm

We therefore seek an approximation to cater to large problems. Non-metric UFLP has a greedy solution [Hochbaum, 1982] with an approximation ratio of $1 + \log(n)$ based on a mapping to Minimum Weight Set Cover (MWSC). Our problem is different from UFLP in several respects, chiefly the aspect demands, precluding direct reuse of that particular greedy solution. Even when confined to one aspect, there is no existing solution with provable guarantee for MWSC with constraint on the number of sets [Golab *et al.*, 2015].

Our proposed greedy solution is Algorithm 1. Sentences in $\mathcal{S}_j$ are the coverable elements. A covering set is a review $t_{i'j}$ with its selected sentences $\tau_{i'}$ to cover a subset of S; its weight is

$$\frac{\theta_{i'} + \sum_{s \in \tau_{i'}} \sum_{s' \in S} \delta_{ss'} \cdot \Gamma_{ss'}}{\sum_{s \in \tau_{i'}} \sum_{s' \in S} \Gamma_{ss'}}$$

Enumerating all subsets is exponential. In practice, it is sufficient to sort $s' \in S$ in terms of $\delta_{ss'}$ and investigate the first k sentences for various k [Hochbaum, 1982]. We greedily pick the lowest-weight set until all the sentences are covered.

Unique to our scenario is the selection of $\tau_{i'}$ from the sentences in $t_{i'j}$, by maximizing the representation of aspects, which always lowers the cost of representation. If there are multiple sentences that can represent an aspect, we seek the permutation with the lowest cost. To ensure coverage, the last sentence should cover all remaining sentences of the aspect.

Complexity Analysis. In Algorithm 1, the two outer loops (lines 2–3) may require $O(|\mathcal{S}_j| \cdot |\mathcal{T}_j|)$. The inner cost is dominated by line 4. Computing the cost is $O(t_{avg} \cdot |\mathcal{S}_j|)$, where t_{avg} is the average length of reviews. Sorting the covered sentences is $O(|\mathcal{S}_j| \log |\mathcal{S}_j|)$. Since $t_{avg} \cdot |\mathcal{T}_j|$ is equivalent to $|\mathcal{S}_j|$, the overall complexity of SEER-Greedy is $O(|\mathcal{S}_j|^3 + |\mathcal{T}_j||\mathcal{S}_j|^2 \log |\mathcal{S}_j|)$.

4 Opinion Contextualization

The goal is an explanation with compatible opinions to the ones the target user would have (as encoded in the Z). Contextualizing the sentences to fit the target user's aspect sentiments is done via two complementary mechanisms.

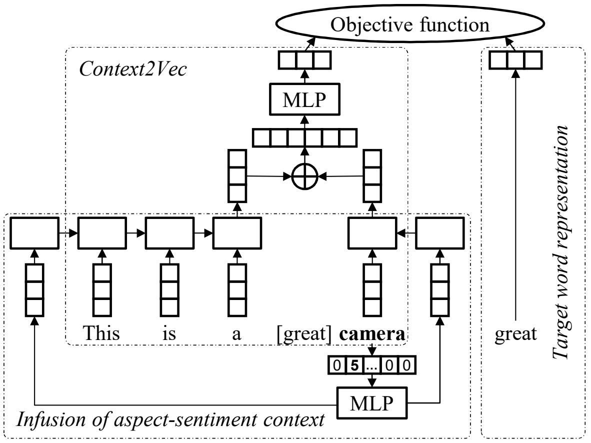

Figure 2: ASC2V Architecture

Sentence Selection. One means is to employ $\theta_{i'}$ that favors more compatible reviewers in Equation 2. $\theta_{i'}$ is defined as a function of the similarity between $z_{ij:}$ (a vector of aspect-level sentiments by target user u_i on p_j) and $z_{i'j:}$, e.g.,

$$\theta_{i'} = \frac{1 - cos(z_{ij:}, z_{i'j:})}{2}$$

Alternatively, our framework could admit other definitions for $\theta_{i'}$ as well.

Opinion Substitution. To "extend" beyond the original pool of review sentences, we contextualize candidate sentences by allowing substitution of the original opinion phrase with another more attuned to the target user's sentiments. After removing the opinion to be substituted, this turns into a sentence completion task, which is an NLP problem in its own right. For concreteness, we allude to a specific solution, but a fuller consideration is beyond the scope of this work. *Context2Vec* [Melamud *et al.*, 2016] pays attention to the entire sentential context, with two LSTMs for sentence-level representation: one reads from the left (lLS) and the other from the right (rLS). Their concatenation passes through a 2-layer perceptron with ReLU activation to get its context representation. L_1, L_2 are fully connected linear operations.

$$\vec{w_l} = L_2(ReLU(L_1(\text{lLS}(w_{1:l-1}) \oplus \text{rLS}(w_{|s|:l+1}))))$$

As *Context2Vec* only considers the surrounding words, the sentence completion is irrespective of the user's aspect-level sentiment. To "personalize" the explanation, we use our modification, called *Aspect-Sentiment Context2Vec* or *ASC2V*, for predicting opinionated word based on sentence context, and z_{ijk}, i.e., u_i's sentiment for aspect a_k of p_j. To infuse this information explicitly, we construct an aspect-sentiment vector $\vec{as}$ of dimensionality $|\mathcal{A}|$. If the sentence is of aspect a_k, we set the k^{th} dimension to the value of z_{ijk}, and 0 otherwise. We use 1-layer perceptron with tanh activation to project aspect sentiment information into the same space as context word embedding. L_3 is fully connected linear operation.

$$\vec{w} = tanh(L_3(\vec{as}))$$

This $\vec{w}$ is the starting token for both lLS and rLS (see Figure 2). We rank candidate opinions based on cosine similarity of their embeddings with the context vector. For the example "This is a ___ camera", if z_{ijk} expresses positive sentiment, "great" should be ranked highly. If negative, a different opinion may apply.

Algorithm 2 Opinion Substitution

```
1: Initialize $min_{\text{r\_cost}} := \text{r\_cost}(\tau)$
2: for $s \in \tau$ do
3:     $\tau' := \tau \backslash \{s\}$; $w_{\text{best}} := \text{get\_opinion}(s)$
4:     for $w \in O_{ijs}$ do
5:         $current\_cost := \text{r\_cost}(\tau' \cup \{s(w)\})$
6:         if $current\_cost < min_{\text{r\_cost}}$ then
7:             $w_{\text{best}} := w$; $min_{\text{r\_cost}} := current\_cost$
8:     $\tau := (\tau \backslash \{s\}) \cup \{s(w_{\text{best}})\}$
9: return $\tau$
```

ASC2V contextualizes sentences within the synthesized explanation to further improve the objective in Equation 3. Let O_{ijs} be top-k predicted opinions for sentence s based on *ASC2V* (for experiments, we use $k = 10$). As shown in Algorithm 2, we substitute each opinion phrase $w \in O_{ijs}$ (line 4) into s by $s(w)$ and keep the one minimizing r_cost (line 7). c_cost is not affected as only the opinion, but not the sentence, changes. This computation is efficient at $O(|\tau| \cdot k)$, as the solution size $|\tau|$ and number of opinions k are usually relatively small.

5 Related Work

5.1 Compatible Recommendation Models

The class of compatible models are broadly defined. [Zhang *et al.*, 2014; Bauman *et al.*, 2017] are based on matrix factorization, while [Chen *et al.*, 2016; Wang *et al.*, 2018a] are based on tensor factorization. Others combine matrix factorization with topic modeling [Wu and Ester, 2015]. Several works enhance their explainable models by using graphs [He *et al.*, 2015] or trees [Gao *et al.*, 2019]. As concrete examples, in Section 6, we experiment with two models, EFM and MTER, which were established methods for templated explanations.

Explicit Factor Model or *EFM* [Zhang *et al.*, 2014] reconstructs the observed rating matrix R, user attention matrix X, and product quality matrix Y. Each $x_{ik} \in X$ indicates the importance of aspect a_k to user u_i, while each $y_{jk} \in Y$ is the summative quality of product p_j on aspect a_k. EFM decomposes the observations X, Y, and R into latent factors, minimizing the function

$$||PQ^T - R||_F^2 + \lambda_x||\eta_1\psi^T - X||_F^2 + \lambda_y||\eta_2\psi^T - Y||_F^2$$

where $P = [\eta_1\ \phi_1]$ and $Q = [\eta_2\ \phi_2]$ are users' and products' latent factors respectively. Each is the concatenation of aspect-based factors (η_1, η_2) influenced by X, Y and hidden factors (ϕ_1, ϕ_2) influenced by ratings. ψ are the latent factors of aspects. Coefficients λ_x and λ_y weigh the relative importance of aspects vs. ratings. We derive Z from the Hadamard product of the reconstructions $\hat{X}, \hat{Y}$, i.e., $z_{ijk} = \hat{x}_{ik} \times \hat{y}_{jk}$.

Multi-Task Explainable Recommendation or *MTER* [Wang *et al.*, 2018a] models user-product-aspect interactions jointly

Dataset	#User	#Product	#Aspect	#Opinion	#Review	#Sentence
Computer	19,818	8,606	5,354	4,243	163,894	512,703
Camera	4,770	2,680	2,321	2,367	37,856	151,382
Toy	2,672	1,984	818	1,225	26,598	57,260
Cellphone	2,340	1,390	882	1,256	19,109	51,469

Table 2: Data statistics

as a tensor G, where $g_{ijk} \in G$ reflects the aggregate sentiment scores across all mentions by user u_i of aspect a_k in product p_j's reviews. The rating r_{ij} is appended as an additional aspect to the tensor G, i.e., $g_{ijv} = r_{ij}$. G is decomposed using Tucker decomposition [Kolda and Bader, 2009]. Let $\hat{G}$ be its reconstruction after minimizing the function

$$||\hat{G} - G||_F - \lambda \sum_{u_i \in \mathcal{U}} \sum_{(u_i, p_j, p'_j)} \ln \sigma(\hat{g}_{ijv} - \hat{g}_{ij'v})$$

where (u_i, p_j, p'_j) is a pairwise ranking observation where u_i prefers p_j to $p_{j'}$. We synthesize an explanation based on the non-rating aspects of $\hat{G}$, i.e., $z_{ij(0:v-1)} = \hat{g}_{ij(0:v-1)}$.

5.2 Comparable Methods

Our baselines comprise methods that could still serve the purpose of recommendation explanation despite not having been designed specifically for that. For one, we could select whole review(s) as explanations. [Tsaparas *et al.*, 2011] selects a set of reviews that maximize the coverage of a specified list of aspects. [Lappas *et al.*, 2012] finds a characteristic set of reviews that best mirror the global distribution of sentiments in the corpus.

We could employ extractive summarization [Barrios *et al.*, 2016] that combines sentences from reviews based on representativeness objective. However, our problem is distinct in incorporating a target user's "would-be"' aspect sentiments in arriving at an explanation with compatible sentiments.

For text generation, recent works utilize LSTM with attention. [Dong *et al.*, 2017] takes into account the user, item, and given rating. [Ni and McAuley, 2018] incorporates the user and item, as well as starter phrases. [Ni *et al.*, 2019] uses history reviews and keywords as attributes. Our synthesis approach selects human-created sentences, rather than generate sentences from abstract representations.

Other methods address different problems and are not comparable. [Li *et al.*, 2017] conditions review generation on latent factors, [Lu *et al.*, 2018] extends on review textual features, while [Truong and Lauw, 2019] extends on images, whereas [Chen *et al.*, 2019] conditions on aspects. [Wang *et al.*, 2018b] applies reinforcement approach for selecting sentences that agree with predicting ratings. Synthetic reviews were considered for unrelated applications, such as simulating spam [Sun *et al.*, 2013].

6 Experiments

Comparisons are tested with one-tailed paired-sample Student's t-test at 0.05 level. Experiments were run on machine with Intel Xeon E5-2650v4 2.20 GHz CPU and 256GB RAM.

Dataset	EFM				MTER			
	Coverage	Overall Cost	Solve Time	# Optimal Solution	Coverage	Overall Cost	Solve Time	# Optimal Solution
Computer	100.00	100.83	4.37	95.07	100.00	100.85	4.05	95.11
Camera	100.00	100.98	4.07	95.55	100.00	100.78	3.67	95.64
Toy	100.00	100.62	2.86	99.95	100.00	100.11	2.19	99.95
Cellphone	100.00	100.81	3.79	98.05	100.00	100.31	3.12	98.05
Total	100.00	100.84	4.16	95.73	100.00	100.74	3.78	95.77

Table 3: Performance ratios of SEER-Greedy to SEER-ILP (%)

		Computer	Camera	Toy	Cellphone
EFM	SEER$_{\text{tfidf}}$	**15.14**§	**14.74**§	**16.36**§	**14.96**§
	SEER$_{\text{SSE}}$	14.48	14.01	15.39	14.40
	SEER$_{\text{ESIM}}$	13.80	13.51	14.81	14.10
MTER	SEER$_{\text{tfidf}}$	**15.15**§	**14.71**§	**16.28**§	**15.03**§
	SEER$_{\text{SSE}}$	14.49	14.03	15.37	14.42
	SEER$_{\text{ESIM}}$	13.79	13.52	14.84	14.10

§ denotes statistically significant improvements. Highest values are in **bold**

Table 4: Comparison of representative costs: ROUGE-L

Datasets. Experiments use four public datasets of Amazon reviews[1] [McAuley *et al.*, 2015] of varying categories: *Computer and Accessories* (Computer), *Camera and Photo* (Camera), *Toys and Games* (Toy), *Cell Phones and Accessories* (Cellphone). Preprocessing follows [Wang *et al.*, 2018a]. For each category, we retrieve the most common aspects covering 90% of opinion phrases and filter out users and items with fewer than five reviews. The remaining are split into training, validation, and test at a ratio of $0.6 : 0.2 : 0.2$ for every user chronologically. Sentences in validation and test with opinions or aspects that had not appeared in training were excluded. Table 2 shows some basic statistics of the datasets.

Base Models. SEER uses aspect-level sentiments Z from two compatible explainable recommendation model (see Section 5). For EFM[2], as in the original work, the latent factor and explicit factor dimensions are 60 and 40. For MTER, we adopt the default setting of the author's implementation[3]. It is not our intention to compare these two, as our model works with any compatible base recommendation method.

Evaluation Metrics. We use ROUGE [Lin and Hovy, 2003], a well-known metric for text matching and text summarization, to assess how well the synthesized explanations approach the ground-truth reviews. To cater to words as well as phrases, we report ROUGE-1 (1-gram) as well as ROUGE-L (longest common subsequence) summatively in terms of the F-Measure.

6.1 Explanation Synthesis

Optimal vs. Approximation. For the optimal SEER-ILP, within 100 seconds, the CPLEX[4] solver can solve optimally for $\geq$ 95% of problem instances. Running on the same instances, SEER-Greedy achieves identical coverage of aspects (100%) at an overall cost that is just 1% higher than optimal, yet consumes merely 4% (i.e., a couple of seconds) of the

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

[2] https://github.com/PreferredAI/cornac

[3] https://github.com/MyTHWN/MTER

[4] https://www.ibm.com/analytics/cplex-optimizer

Model	Computer		Camera		Toy		Cellphone	
	MRR	R@10	MRR	R@10	MRR	R@10	MRR	R@10
C2V	0.460	0.695	0.411	0.645	0.515	0.705	0.365	0.621
RC2V	0.462	0.706	0.409	0.643	0.514	0.707	0.366	0.624
ASC2V$_{\text{EFM}}$	**0.475**§	**0.713**§	**0.416**§	**0.652**§	**0.526**§	**0.726**§	**0.384**§	**0.649**§
ASC2V$_{\text{MTER}}$	**0.473**§	**0.709**§	**0.418**§	**0.653**§	**0.528**§	**0.724**§	**0.388**§	**0.651**§

§ denotes statistically significant improvements by ASC2V
Highest values (among ASC2V, RC2V, and C2V) are in **bold**

Table 5: Opinion Contextualization

time taken by SEER-ILP on average (see Table 3). Subsequently, we run both variants on 100% of the problem instances. For ILP, the result would reflect either the optimal or the best solution up to that point.

Representativeness Cost. For the representativeness cost $\delta_{ss'}$ in Equation 1, we explore several options. One is based on the cosine similarity of sentences s and s'. Each sentence is represented by *tfidf* vectors based on the vocabulary of product's sentences. For $\delta_{ss'}$, we take

$$\delta_{ss'} = \frac{1 - cos(s, s')}{2}$$

We also try two other models: SSE [Nie and Bansal, 2017] for paraphrase identification and ESIM [Chen *et al.*, 2017] for textual entailment. Table 4 shows *tfidf* to perform the best in terms of ROUGE-L. We will use it subsequently. One reason is the corpus SSE and ESIM trained on was not optimized for review sentences. In any case, we consider $\delta_{ss'}$ as given.

6.2 Opinion Contextualization

We hide the ground-truth opinion from the held-out test review and evaluate the ranking of candidates in $\mathcal{O}$ using IR metrics: MRR (the reciprocal rank of the true opinion, averaged across held-out reviews) and Recall@10 or R@10 (fraction of held-out reviews with the true opinion in the top-10).

We compare our ASC2V with two baselines. *Context2Vec* or C2V [Melamud *et al.*, 2016] with only on the sentence (no aspect sentiment). RC2V uses random aspect sentiment. For ASC2V, we train with similar setting as C2V, using RMSprop for optimization. Table 5 shows both variants of ASC2V significantly outperform C2V. RC2V, which adds no meaningful information, fluctuates around C2V. Indeed aspect-level sentiments are useful for opinion contextualization.

6.3 Comparison to Baselines

We compare the explanations generated by SEER to several categories of baselines. For parity, we control for the explanation length. The first category is *template explanation*, comprising the original explanations by EFM [Zhang *et al.*, 2014] and MTER [Wang *et al.*, 2018a]. Next is *review summarization* represented by TEXT RANK [Barrios *et al.*, 2016] and *review selection* with four methods: REPRESENTATIVE selects the review with lowest representative cost (see Equation 1); COMPREHENSIVE selects the review of highest aspect coverage [Tsaparas *et al.*, 2011]; CHARACTERISTIC selects the review whose aspect sentiment distribution most resembles a product's reviews [Lappas *et al.*, 2012]; CHARACTERISTIC+ that also takes into account the aspect demand by considering distributions of demanded aspects only. The last category is *review generation* with ATT2SEQ [Dong *et al.*, 2017] that generates text from user, product, and rating as attributes; EXPANSION NET [Ni and McAuley, 2018] that generates text from aspect words as starter phrases; and AP-REF2SEQ [Ni *et al.*, 2019] that generates text from user & item reviews and aspect words.

	Model	Computer	Camera	Toy	Cellphone
	ATT2SEQ	0.192	0.162	0.257	0.195
	EXPANSION NET	0.478	0.612	0.734	0.504
	AP-REF2SEQ	0.212	0.242	0.367	0.242
	TEXT RANK	0.234	0.219	0.311	0.266
	REPRESENTATIVE	0.408	0.407	0.480	0.448
	COMPREHENSIVE	0.678	0.629	0.717	0.678
	CHARACTERISTIC	0.153	0.169	0.291	0.207
	CHARACTERISTIC+	0.574	0.521	0.662	0.582
EFM	TEMPLATE	0.697	0.654	0.725	0.687
	SEER-Greedy	**0.775**§	**0.729**§	**0.787**§	**0.768**§
	SEER-ILP	**0.775**§	**0.729**§	**0.787**§	**0.768**§
MTER	TEMPLATE	**0.775**§	**0.729**§	**0.787**§	**0.768**§
	SEER-Greedy	**0.775**§	**0.729**§	**0.787**§	**0.768**§
	SEER-ILP	**0.775**§	**0.729**§	**0.787**§	**0.768**§

§ denotes statistically significant improvements. Highest values are in **bold**

Table 6: Comparison to Baselines: Coverage

		Computer		Camera		Toy		Cellphone	
	Model	R-1	R-L	R-1	R-L	R-1	R-L	R-1	R-L
	ATT2SEQ	16.69	10.35	15.90	9.13	16.51	10.41	16.42	9.76
	EXPANSION NET	11.68	1.25	19.23	5.19	24.41	4.68	14.13	3.11
	AP-REF2SEQ	16.94	12.29	17.04	12.94	21.72	14.50	19.15	12.99
	TEXT RANK	18.68	11.15	19.29	11.37	19.04	11.97	19.16	11.25
	REPRESENTATIVE	18.22	11.11	19.24	11.27	19.72	12.60	19.45	11.80
	COMPREHENSIVE	21.90	13.44	22.16	13.16	23.41	15.12	22.33	13.73
	CHARACTERISTIC	13.18	7.65	14.05	7.92	15.76	9.80	14.27	8.41
	CHARACTERISTIC+	18.33	10.87	18.06	10.32	21.25	13.56	19.05	11.34
EFM	TEMPLATE	14.17	8.41	14.43	8.39	13.37	8.06	14.22	8.41
	SEER-Greedy	24.89§	15.05§	25.11§	14.72§	25.33§	16.30§	24.66§	14.87§
	SEER-ILP	**25.12**§	15.14§	**25.23**§	**14.74**§	**25.43**§	**16.36**§	**24.76**§	14.96§
MTER	TEMPLATE	16.88	11.68	16.43	11.14	13.22	12.03	17.61	11.86
	SEER-Greedy	24.90§	15.08§	25.01§	14.65§	25.25§	16.24§	24.74§	14.94§
	SEER-ILP	**25.12**§	**15.15**§	25.22§	14.71§	25.33§	16.28§	24.85§	**15.03**§

§ denotes statistically significant improvements by our models
Highest values in **bold**

Table 7: Comparison to Baselines: ROUGE-1 and ROUGE-L

Coverage. Table 6 shows the coverage, i.e., the proportion of the met aspect demand. Coverage is not necessarily 1 due to the limited number of candidate sentences for selection or aspects that have not appeared before. Both SEER variants outperform baselines in coverage (MTER has identical coverage). The template methods respond to aspect demand. EFM produces duplicate sentences for an aspect, resulting in lower coverage than MTER that produces multiple sentences by varying opinion phrases. Methods that do not benefit from aspect demands (including ATT2SEQ, TEXT RANK, REPRESENTATIVE, and CHARACTERISTIC) underperform the other methods that do. Review selection methods are limited to what can be covered by a review. Among these, COMPREHENSIVE achieves the highest aspect coverage. As the review with the closest aspect sentiment distribution does not necessarily have the highest aspect coverage, the coverage of CHARACTERISTIC+ is lower than COMPREHENSIVE.

Ground Truth Recovery. As Table 7 shows, SEER variants (ILP and Greedy) significantly outperform all the baselines, with the highest F-Measure for both ROUGE-1 (R-1)

User		A3ALXLASGICTBU
Product		B002DPUUKK
Title		Microsoft Wireless Mobile Mouse 4000 - White
Ground truth		The mouse has worked great for about 1-year. The mouse was great for a while. The size is perfect for my hand
Att2Seq		I really like the mouse. The mouse is very comfortable and the mouse is fine. I haven't had any problems with the wireless signal
Expansion Net		The mouse is a plus. The size is great and the size is perfect
AP-Ref2Seq		It's a good wireless mouse for the price. It's a good wireless mouse. It's a good mouse for the price
Text Rank		This is a great mouse. A great mouse. Very good mouse
Representative		If you call up with a problem mouse that requires a replacement. An all black mouse is difficult to find inside a laptop bag in the dark. I selected the "downtown" version with the white glossy center panel and "city grid/skyline" motif
Comprehensive		A great mouse. 4 stars instead of 5 because of the lightness and the smooth mouse wheel instead of the ratcheting one. The size is good
Characteristic		I got this mouse instead of a 3000 series because of the extra button on the side. The side button is not handy because of how it is placed so high and forward on the mouse. In which case you might not mind
Characteristic+		Thinking a wireless mouse would be good
EFM	Template	You might be interested in [mouse], on which this product performs well. You might be interested in [size], on which this product performs well
	SEER-ILP	The <mouse> is very [comfortable] and nice looking. This is a [great] <mouse>. The <size> is [perfect]
MTER	Template	Its <mouse> is [easy-to-adjust]. Its <mouse> is [lefty]. Its <size> is [awkward]
	SEER-ILP	The <mouse> is very [comfortable] and nice looking. This is a [great] <mouse>. The <size> is [good]

Table 8: Example Explanations on a Computer instance

	Model	**Annotator**					**Average**
		1	**2**	**3**	**4**	**5**	
Q1	MTER	2.10	2.35	2.75	3.00	2.35	2.51
	AP-Ref2Seq	3.15	3.00	3.60	3.75	3.50	3.40
	SEER-ILP	**3.95**§	**4.25**§	**4.00**§	**3.85**§	**4.10**§	**3.75**§
Q2	MTER	1.75	1.50	2.40	2.80	2.00	2.09
	AP-Ref2Seq	1.95	3.05	3.20	3.40	3.10	2.94
	SEER-ILP	**3.55**§	**4.45**§	**3.80**§	**3.75**§	**4.45**§	**3.50**§

§ denotes statistically significant improvements. Highest values are in **bold**

Table 9: Result analysis of user study

and ROUGE-L (R-L)[5]. The template-based approaches perform poorly because a standard template cannot reflect varied reviews. Benefitting from paying attention to the aspect demand, Characteristic+ performs better than Characteristic. However, both still perform worse than Comprehensive that maximizes coverage of aspect demand. Representative outperforms Characteristic since it optimizes for representativeness yet is still lower than Comprehensive. Text Rank underperforms Comprehensive, because of redundant sentences that repeat aspects while the latter considers a whole review covering various aspects. The review generation approaches tend to produce short and repetitive sentences. They do not reflect aspect-level sentiments fully: Att2Seq uses ratings but no aspects, whereas Expansion Net and AP-Ref2Seq use aspects but may not reflect sentiments well.

6.4 Qualitative Study

Case Study. As an illustration, Table 8 shows the explanations for a Computer instance. The ground truth review reveals aspect demand involving *mouse* and *size*. EFM describes the product having good performance on the two aspects. MTER opinions are difficult to understand. Att2Seq does not cover the aspect demand. Expansion Net generates short sentences repetitively. Text Rank tends to select popular repetitive aspects. Our SEER-ILP produces readable explanations that reflect not only the aspects, but also the user opinions. When used with EFM or MTER, it generates slightly different phrases, e.g., "perfect" vs. "good" *size*.

User Study. To test the efficacy of the explanations from human perspective, we randomly select 5 user-product pairs from each category to get 20 examples in total and design a survey involving five participants who are not the authors. There are two questions. The first looks into the language quality, e.g., readable and easy to understand. The second, which also appeared in [Wang *et al.*, 2018a], looks into appropriateness of recommendation.

Q1: Are the explanatory sentences well-formed and understandable?

Q2: Does the explanation help you understand why the given product is being recommended to the given user?

Each question is applied to a given explanation. Each participant chooses from five-point Likert scale, from 1 (strongly disagree) to 5 (strongly agree). To compare to the proposed SEER-ILP, we choose MTER and AP-Ref2Seq as representative baselines, as these two were designed specifically for recommendation explanation and achieve high performance in terms of ROUGE-L. As reported in Table 9, SEER-ILP outperforms the two baselines significantly. For Q1, MTER with simple template is difficult to understand, while AP-Ref2Seq achieves better results (≥ 3) comparing to MTER which shows its ability to generate readable text. However, AP-Ref2Seq-generated text is short and too general which make their explanations less informative than those of SEER-ILP.

7 Conclusion

We propose an innovative post hoc strategy for providing natural language explanations for personalized recommendations. Our approach synthesizes an explanation by selecting representative sentences from a product's reviews, contextualizing the opinions based on aspect-level sentiments from a class of compatible explainable recommendation models. SEER performs well against competitive baselines including templates, review summarization, selection, and generation.

Acknowledgments

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its NRF Fellowship Programme (Award No. NRF-NRFF2016-07).

[5] We have experimented with other ROUGE variations (ROUGE-[1,2,L],S[1-4],SU[1-4]) with consistent results. SEER outperforms other baselines significantly in term of F-Measure. For conciseness, we report only the F-measure of ROUGE-1 and ROUGE-L.

References

[Barrios *et al.*, 2016] Federico Barrios, Federico López, Luis Argerich, and Rosa Wachenchauzer. Variations of the similarity function of textrank for automated summarization. *CoRR*, abs/1602.03606, 2016.

[Bauman *et al.*, 2017] Konstantin Bauman, Bing Liu, and Alexander Tuzhilin. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *KDD*, pages 717–725, New York, NY, USA, 2017. ACM.

[Chen *et al.*, 2016] Xu Chen, Zheng Qin, Yongfeng Zhang, and Tao Xu. Learning to rank features for recommendation over multiple categories. In *SIGIR*, SIGIR '16, pages 305–314, New York, NY, USA, 2016. ACM.

[Chen *et al.*, 2017] Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. Enhanced LSTM for natural language inference. In *ACL*, pages 1657–1668, Vancouver, Canada, 2017. ACL.

[Chen *et al.*, 2019] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. Co-attentive multi-task learning for explainable recommendation. In *IJCAI*, pages 2137–2143. AAAI Press, 2019.

[Cornuéjols *et al.*, 1983] Gérard Cornuéjols, George L Nemhauser, and Lairemce A Wolsey. The uncapacitated facility location problem. Technical report, Carnegie-mellon univ pittsburgh pa management sciences research group, 1983.

[Dong *et al.*, 2017] Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. Learning to generate product reviews from attributes. In *EACL*, volume 1, pages 623–632, 2017.

[Gao *et al.*, 2019] Jingyue Gao, Xiting Wang, Yasha Wang, and Xing Xie. Explainable recommendation through attentive multiview learning. In *AAAI*, volume 33, pages 3622–3629, 2019.

[Golab *et al.*, 2015] Lukasz Golab, Flip Korn, Feng Li, Barna Saha, and Divesh Srivastava. Size-constrained weighted set cover. In *ICDE*, pages 879–890. IEEE, 2015.

[He *et al.*, 2015] Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. Trirank: Review-aware explainable recommendation by modeling aspects. In *CIKM*, CIKM '15, pages 1661–1670, New York, NY, USA, 2015. ACM.

[Hochbaum, 1982] Dorit S Hochbaum. Heuristics for the fixed cost median problem. *Math. Program.*, 22(1):148–162, 1982.

[Kolda and Bader, 2009] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[Lan and Xu, 2018] Wuwei Lan and Wei Xu. Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *COLING*, pages 3890–3902, 2018.

[Lappas *et al.*, 2012] Theodoros Lappas, Mark Crovella, and Evimaria Terzi. Selecting a characteristic set of reviews. In *KDD*, pages 832–840. ACM, 2012.

[Li *et al.*, 2017] Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. Neural rating regression with abstractive tips generation for recommendation. In *SIGIR*, page 345–354, New York, NY, USA, 2017. Association for Computing Machinery.

[Lin and Hovy, 2003] Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACL*, pages 71–78, Stroudsburg, PA, USA, 2003. ACL.

[Lu *et al.*, 2018] Yichao Lu, Ruihai Dong, and Barry Smyth. Why i like it: Multi-task learning for recommendation and explanation. In *RecSys*, page 4–12, New York, NY, USA, 2018. ACM.

[Ma *et al.*, 2019] Weizhi Ma, Min Zhang, Yue Cao, Woojeong Jin, Chenyang Wang, Yiqun Liu, Shaoping Ma, and Xiang Ren. Jointly learning explainable rules for recommendation with knowledge graph. In *WWW*, WWW '19, pages 1210–1221, New York, NY, USA, 2019. ACM.

[McAuley *et al.*, 2015] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. Image-based recommendations on styles and substitutes. In *SIGIR*, pages 43–52, New York, NY, USA, 2015. ACM.

[Melamud *et al.*, 2016] Oren Melamud, Jacob Goldberger, and Ido Dagan. context2vec: Learning generic context embedding with bidirectional lstm. In *CoNLL*, pages 51–61, 2016.

[Ni and McAuley, 2018] Jianmo Ni and Julian McAuley. Personalized review generation by expanding phrases and attending on aspect-aware representations. In *ACL*, volume 2, pages 706–711, 2018.

[Ni *et al.*, 2019] Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP*, pages 188–197. ACL, 2019.

[Nie and Bansal, 2017] Yixin Nie and Mohit Bansal. Shortcut-stacked sentence encoders for multi-domain inference. In *RepEval*, pages 41–45, Copenhagen, Denmark, 2017. Association for Computational Linguistics.

[Ren *et al.*, 2017] Zhaochun Ren, Shangsong Liang, Piji Li, Shuaiqiang Wang, and Maarten de Rijke. Social collaborative viewpoint regression with explainable recommendations. In *WSDM*, pages 485–494, New York, NY, USA, 2017. ACM.

[Sun *et al.*, 2013] Huan Sun, Alex Morales, and Xifeng Yan. Synthetic review spamming and defense. In *KDD*, pages 1088–1096. ACM, 2013.

[Tan *et al.*, 2016] Yunzhi Tan, Min Zhang, Yiqun Liu, and Shaoping Ma. Rating-boosted latent topics: Understanding users and items with ratings and reviews. In *IJCAI*, volume 16, pages 2640–2646, 2016.

[Truong and Lauw, 2019] Quoc-Tuan Truong and Hady Lauw. Multimodal review generation for recommender systems. In *WWW*, pages 1864–1874, New York, NY, USA, 2019. ACM.

[Tsaparas *et al.*, 2011] Panayiotis Tsaparas, Alexandros Ntoulas, and Evimaria Terzi. Selecting a comprehensive set of reviews. In *KDD*, pages 168–176, New York, NY, USA, 2011. ACM.

[Wang *et al.*, 2018a] Nan Wang, Hongning Wang, Yiling Jia, and Yue Yin. Explainable recommendation via multi-task learning in opinionated text data. In *SIGIR*, SIGIR '18, pages 165–174, New York, NY, USA, 2018. ACM.

[Wang *et al.*, 2018b] Xiting Wang, Yiru Chen, Jie Yang, Le Wu, Zhengtao Wu, and Xing Xie. A reinforcement learning framework for explainable recommendation. In *ICDM*, pages 587–596. IEEE, 2018.

[Wu and Ester, 2015] Yao Wu and Martin Ester. Flame: A probabilistic model combining aspect based opinion mining and collaborative filtering. In *WSDM*, WSDM '15, pages 199–208, New York, NY, USA, 2015. ACM.

[Zhang *et al.*, 2014] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *SIGIR*, SIGIR '14, pages 83–92, New York, NY, USA, 2014. ACM.