Rationale-based Opinion Summarization

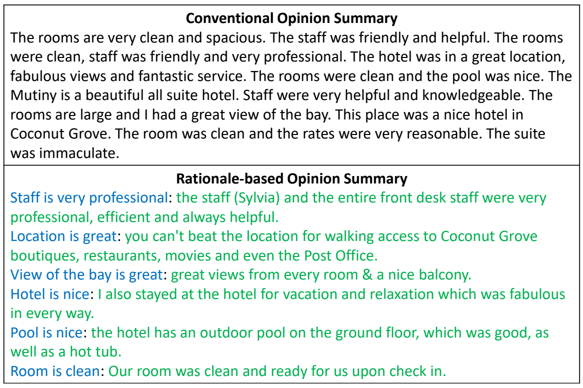
Anonymous ACL submission

Abstract

Opinion summarization aims to generate concise summaries that present popular opinions of a large group of reviews. However, these summaries can be too generic and lack supporting details. To address these issues, we propose a new paradigm for summarizing reviews, rationale-based opinion summarization. Rationale-based opinion summaries outputs the representative opinions as well as one or more corresponding rationales. To extract good rationales, we define four desirable properties: relatedness, specificity, popularity, and diversity and present a Gibbs-sampling-based method to extract rationales. Overall, we propose RATION , an unsupervised extractive system that has two components: an Opinion Extractor (to extract representative opinions) and Rationales Extractor (to extract corresponding rationales). We conduct automatic and human evaluations to show that rationales extracted by RATION have the proposed properties and its summaries are more useful than conventional summaries.

1 Introduction

Online reviews are useful for both customers and businesses (Cheung et al., 2012). However, the large number of reviews on such platforms makes it difficult to manually read all of them. Opinion summarization aims to tackle this problem by generating a concise summary of the reviews. Recently, much progress has been made in opinion summarization, especially unsupervised summarization. These works either extract sentences from reviews as summaries (Angelidis et al., 2021a; Basu Roy Chowdhury et al., 2022; Li et al., 2023) or generate summaries conditioned on reviews (Chu and Liu, 2019; Amplayo and Lapata, 2020). However, such summaries are usually very generic and lack supporting evidence. To address this issue, Suhara et al. (2020); Bar-Haim et al. (2021); Hosking et al. (2023) produce summaries in which the summarizing content is attributed to a group of supporting review sentences. However, since their goal is to explain the choice of the summary content, the sizes of these groups of supporting sentences are too large to be useful for user consumption.



Conventional Opinion Summary

The rooms are very clean and spacious. The staff was friendly and helpful. The rooms were clean, staff was friendly and very professional. The hotel was in a great location, fabulous views and fantastic service. The rooms were clean and the pool was nice. The Mutiny is a beautiful all suite hotel. Staff were very helpful and knowledgeable. The rooms are large and I had a great view of the bay. This place was a nice hotel in Coconut Grove. The room was clean and the rates were very reasonable. The suite was immaculate.

Rationale-based Opinion Summary

Staff is very professional: the staff (Sylvia) and the entire front desk staff were very professional, efficient and always helpful.
Location is great: you can't beat the location for walking access to Coconut Grove boutiques, restaurants, movies and even the Post Office.
View of the bay is great: great views from every room & a nice balcony.
Hotel is nice: I also stayed at the hotel for vacation and relaxation which was fabulous in every way.
Pool is nice: the hotel has an outdoor pool on the ground floor, which was good, as well as a hot tub.
Room is clean: Our room was clean and ready for us upon check in.

Figure 1: Examples of a conventional and a rationale-based opinion summary (generated by RATION) for the same entity. In rationale-based summary, each line presents a representative opinion and its rationale.

In this paper, we propose a new paradigm for summarizing reviews, rationale-based opinion summarization. Given a set of reviews about an entity (such as a hotel), rationale-based opinion summarization outputs *representative opinions* summarizing the reviews as well as one or more *rationales* for each representative opinion. Fig. 1 shows an example of a conventional summary produced by a recent extractive summarization model (top) and a rationale-based summary (bottom) containing representative opinions (in blue) and corresponding rationales (in green) for the same entity, a hotel in this case. For illustration, we show only one rationale per representative opinion in the figure but in practice, there can be several such rationales specified by users. Such rationale-based summaries can be more useful to users by providing representative opinions as well as informative rationales for them, helping users in making decisions.

Rationale-based opinion summarization presents several major challenges: (i) what makes a good rationale? and (ii) how to extract rationales? To address the first challenge, we define four desirable properties for rationales: **relatedness**, **specificity**, **popularity**, and **diversity**. To address the second challenge, we present methods to estimate these properties for review sentences and a Gibbs-sampling-based approach to extract review sentences that can serve as rationales.

Overall, we propose RATION (see Fig. 2), an unsupervised extractive system that has two components: an *Opinion Extractor* (to extract representative opinions) and a *Rationales Extractor* (to extract corresponding rationales). Both the representative opinions and corresponding rationales are extracted from the input review sentences in an unsupervised manner and are presented together as the final output summary. The Opinion Extractor extracts representative opinions about various aspects of the entity in a concise manner and removes redundancy in them through a graph-based approach. The Rationales Extractor first estimates the four above-mentioned properties of good rationales. Since there is no supervision in the review domain for estimating some of these properties, RATION uses an alignment model fine-tuned to the domain of reviews using artificially constructed samples. The values of these properties collectively represent the joint probability of a set of review sentence to serve as rationales. For each representative opinion, RATION uses Gibbs Sampling to sample a user-specified number of sentences as rationales by approximating this joint probability distribution.

Our experiments show that rationale-based opinion summaries generated by RATION are more informative and useful than conventional summaries and the rationales generated by RATION are better than those generated by strong baselines. Our contributions are three-fold: Our contributions are:

- We propose a new paradigm for summarizing reviews, rationale-based opinion summarization;
- We design RATION , a model to extract representative opinions and corresponding rationales;
- We evaluate RATION using automatic metrics and human evaluation and show that it outperforms strong baselines.

2 Related Work

There are generally two types of opinion summarization: abstractive and extractive.For abstractive summarization, previous works either use aggregate review sentence representations (Chu and Liu, 2019; Isonuma et al., 2021) or generate synthetic datasets to train generation models in a supervised setting (Bražinskas et al., 2019; Amplayo and Lapata, 2020). For extractive summarization, previous works generally predict the salience of review sentences based on their distance from the aspect representation (Angelidis et al., 2021a), from the average sentence representation (Basu Roy Chowdhury et al., 2022) or from the aspect cluster centers (Li et al., 2023) and extract salient sentences as summaries. However, opinion summaries generated by previous works are usually generic and lack supporting evidence.

To generate more specific opinion summaries, (Iso et al., 2021) generates summaries based on the convex aggregation of review sentence representations instead of the average. However, such summaries might still lack supporting evidence. For explainability, Suhara et al. (2020) cluster the opinions extracted from review sentences and generate summaries based on the clusters. Bar-Haim et al. (2021) matches review sentences to *key points* and extracts the key points that are matched by most review sentences as summaries. Hosking et al. (2023) generates path representation for each review sentence and generates summaries based on the selected paths. These works can attribute their summary content to a group of review sentences. However, since their goal is to explain the choice of the summary content, the sizes of these groups of supporting sentences are too large to be useful for user consumption. RATION aims to address this issue by generating rationale-based opinion summarization where each opinion is supported by a small group of rationales.

3 Problem Statement

The input in rationale-based opinion summarization is a set of review sentences $S = \{s_1, ..., s_n\}$ of a given entity, such as a hotel. The output is a summary D that consists of representative opinions $O = \{o_1, ..., o_m\}$ and corresponding sets of rationales $\mathcal{R} = \{R_1, R_2 ... R_m\}$, where $R_i = \{r_{i,1}, r_{i,2} ... r_{i,k}\}$, $r_{i,*}$ is a rationale, and k is specified by the user. See Fig. 1 for examples of representative opinions and rationales.

4 RATION

RATION addresses this problem using two components: an *Opinion Extractor* (§4.1) and a *Ratio-*

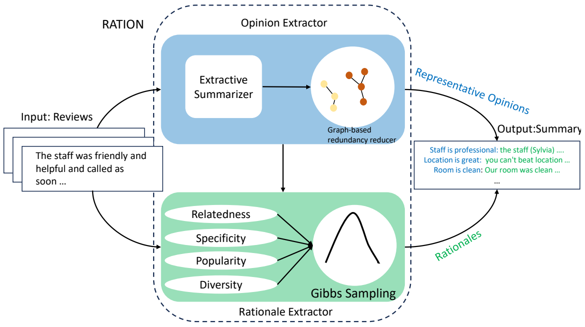

RATION
Opinion Extractor
Extractive Summarizer
Graph-based redundancy reducer
Input: Reviews
The staff was friendly and helpful and called as soon ...
Representative Opinions
Output:Summary
Staff is professional: the staff (Sylvia) ....
Location is great: you can't beat location ...
Room is clean: Our room was clean ...
...
Relatedness
Specificity
Popularity
Diversity
Gibbs Sampling
Rationales
Rationale Extractor

Figure 2: Overview of RATION and its two components: the Opinion Extractor and the Rationales Extractor.

nales Extractor (§4.2). The representative opinions and rationales are extracted from the input review sentences in an unsupervised manner. They are combined to form a summary, D (§4.3). RATION uses an alignment model in its processing which is described in §4.4.

4.1 Opinion Extractor

In this section, we describe how RATION extracts representative opinions O from input review sentences S. Representative opinions should be concise sentences that summarizes the reviewers' impressions of the entity. Since existing summarization models are good at identifying this information, RATION uses an existing extractive opinion summarization model to extract summarizing review sentences. Fig. 1 (top) shows an example.

From these summary sentences, RATION extracts representative opinions of the form 'A is B'. For example, from the review sentence, 'The hotel was in a great location, fabulous views, and fantastic service.', one representative opinion extracted by RATION is 'location is great'. We chose this format because it is concise yet informative. For extracting representative opinions from sentences, RATION uses the model proposed by Miao et al. (2020) that was trained on the ABSA dataset (Miao et al., 2020; Cai et al., 2021). The ABSA dataset consists of review sentences like 'Staff at the hotel is helpful.' annotated with the aspect the sentence is talking about ('service' in this example), sentiment ('positive'), and an opinion ('helpful'). To avoid confusion with our representative opinions, we refer to these opinions as ABSA-opinions. The model we used takes as input a sentence and outputs aspects, sentiments, and ABSA-opinions.

However, since the extracted summarizing sentences are often repetitive, many extracted representative opinions are similar to each other, like 'room is spacious' and 'room is large'. RATION removes the redundancy among the extracted representative opinions based on their relationship with review sentences. It assumes that if two representative opinions are related to a similar group of review sentences, they are likely to be similar. For this, it first estimates the relatedness between a representative opinion o and review sentence s, using an alignment model M_{align} (described in detail later in §4.4). RATION uses the probability $p_{align}(s, o)$ estimated by M_{align} that s aligns with o as the relatedness. Next, using this relatedness, RATION estimates the similarity between two representative opinions o and o'. For this, it constructs a feature vector for every representative opinion, o, $f_o \in R^n$ whose i-th element is $p_{align}(s_i, o)$ if review sentence s_i aligns with o, otherwise it is zero. The similarity between two representative opinions o and o', is defined as the cosine similarity between their feature vectors f_o and $f_{o'}$. Next, to cluster similar representative opinions together, RATION constructs an undirected graph where each node is a representative opinion and there is an edge between two nodes if their similarity is greater than a threshold β. Each connected component of the graph forms an *opinion cluster* G and its most prototypical node (the node that is aligns with the most review sentences) is extracted as a representative opinion $o_i \in O$. The number of representative opinions in O is equal to the number of clusters identified above.

4.2 Rationales Extractor

In this section, we describe how for each representative opinion o_i, RATION extracts a set of k rationales, R_i, from the input review sentences S.

For a given representative opinion, o_i, not all review sentences are viable candidates for its rationales since they might not be relevant to it. We filter out such nonviable candidates and retain only viable ones as the *rationale candidate set* C_i using the alignment model, M_{align}. Let G_i represent the opinion cluster that representative opinion o_i belongs to. A review sentence, s, is included in the candidate set C_i if (i) it aligns with at least one opinion in G_i, and (ii) it is most related to G_i among all clusters. RATION defines the relatedness between review sentence s and cluster G as the maximum alignment score between s and any element of G:

$$e(s, G) = max_{o \in G} p_{ent}(s, o) \tag{1}$$

After removing nonviable candidates, RATION extracts rationales, R_i, from the rationale candidate set, C_i, for each representative opinion o_i. Good rationales should be related to the corresponding representative opinion (relatedness). They should contain specific details (specificity), represent popular information (popularity), and offer diverse information (diversity). We now describe how to quantify these properties and then describe how to extract rationales based on these properties.

Relatedness of review sentence s to representative opinion o_i, $(rel(s))$, measures how related s is to o_i as compared to all other representative opinions. As before, let G_i represent the cluster that o_i belongs to. Using the definition of relatedness between a review sentence s and a cluster G (Equation 1), $rel(s)$ is defined as:

$$rel(s) = \frac{e(s, G_i)}{\sum_{G_k \in G_s} e(s, G_k)} \tag{2}$$

where G_s is the set of the opinion clusters that has at least one element that sentence s aligns with.

Specificity of review sentence s, $(spec(s))$, measures the amount of details that s contains. For this, it uses a Deberta (He et al., 2020) model finetuned on a specificity estimation dataset (Ko et al., 2019).

Popularity of review sentence s, $(pop(s))$, measures how representative it is of the rationale candidate set it belongs to. To calculate $pop(s)$, RATION constructs a weighted undirected graph. The nodes of this graph represent the review sentences in the rationale candidate set C_i, $s \in C_i$. The representative opinion o_i also forms a node. There is an edge between two review sentences if one aligns with the other or vice versa (as estimated by M_{align}). The weight of this edge is the greater of the two alignment probabilities. There is an edge between a review sentence and the representative opinion if the review sentence aligns with the representative opinion and the weight of this edge is the alignment probability. RATION measures the popularity $pop(s)$ of sentence s as the centrality of the corresponding node in this graph.

Diversity of a group of review sentences, $s_{1:k}$, $(div(s_{1:k}))$, measures how dissimilar their content collectively is. It is estimated as the negative of the pairwise cosine similarity of their bag-of-word representations.

Gibbs Rationale Sampler: Based on the properties defined above, RATION defines the joint probability of a group of review sentences, $s_{1:k}$, to be selected as rationales to be proportional to:

$$exp(\sum_{i=1}^{k} sal(s_i) + \gamma div(s_{1:k})) \tag{3}$$

where $\gamma > 0$ is the weight of the diversity term and $sal(s)$ is the product of $rel(s)$, $spec(s)$ and $pop(s)$, each normalized to $[0, 1]$ using min-max normalization among the rationale candidate set s belongs to.

However, directly computing this probability for all possible groups is computationally expensive. To address this issue, RATION uses Gibbs Sampling. Gibbs Sampling is a Markov chain Monte Carlo algorithm that can approximate the joint probability of a group of sentences $s_{1:k} \subset C_i$ being considered as rationales, $R_i = \{r_{i1}, r_{i2} ... r_{ik}\}$, for the representative opinion, o_i. Since the joint probability is difficult to sample from, it iteratively samples individual r_{i*} conditioned on the values of other r_{i*}s. The sequence of samples hence obtained form a Markov chain and its stationary distribution approximates the joint distribution. Using $R_{i\neg j}$ to refer to all elements of R_i except the j^{th} element r_{ij}, the conditional probability $p(r_{ij} = s^* | R_{i\neg j})$ is proportional to:

$$\frac{exp(sal(s^*) + \gamma div(\{R_{i\neg j}, s^*\}))}{\sum_{s \in C_i} exp(sal(s) + \gamma div(\{R_{i\neg j}, s\}))} \tag{4}$$

This sampling process is detailed in Alg. 1. The input of the algorithm is the representative opinion o_i, its rationale candidate set C_i,and η, θ (Line 1). Initially, R_i are randomly sampled from rationale candidate set C_i (Line 2). In each Gibbs update, $r_{i\cdot}$ is sampled from the conditional distribution conditioned on other sentences, $R_{i\neg j}$(Line 6). After the *burn-in period* of η, RATION records the frequency of sampled review sentence group in additional θ scans as R_i to approach the stationary distribution more closely (Line 9). RATION extracts the most frequent review sentence group as the rationales R_i (Line 12).

4.3 Summarization

We now describe how RATION generates summary D using representative opinions O and rationales $\mathcal{R}$. In principle, RATION can simply pair each $o_i \in O$ with the rationales in corresponding $R_i \in \mathcal{R}$. However, sometimes the user might want to put restrictions on the length of the summary. In such cases, RATION gives more importance to representative opinions supported by more review sentences. It obtains them by ranking the representative opinions in O in descending order of the size of the

Algorithm 1 Gibbs Rationale Sampler

```
Input: $\eta$, $\theta$, $o_i$, $C_i$
Randomly initialize $R_i$ from $C_i$
R={}        ▷ R records the frequency of sentence groups
for $l = 1$ to $\eta + \theta$ do
    for $j = 1$ to $k$ do
        sample $r_{ij} \sim p(r_{ij} = s^*|R_{i\neg j})$
        if $l > \eta$ then
            R[$R_i$]+=1
        end if
    end for
end for
$R_i$=$\arg\max'_R$R[$R'$]
return $R_i$
```

corresponding rationale candidate sets. RATION then constructs the summary, D, by picking representative opinions o_i from this ranked list and the corresponding rationales R_i until the length limit is reached (examples shown in Appendix Fig. 4).

4.4 The Alignment Model

At various stages in its processing, RATION uses an alignment model M_{align} to estimate alignment or relatedness between pairs of sentences. M_{align} takes a pair of sentences $\langle$ X, Y$\rangle$ as input, and predicts whether X aligns with Y (*alignment*), X opposes Y (*opposite*) or X is neutral to Y (*neutral*). However, there is no in-domain supervision available for finetuning this alignment model. RATION therefore finetunes a RoBerta (Radford et al., 2019) model on artificially generated samples from the ABSA dataset (described in §4.1). It generates two types of fine-tuning samples: *Sent-Opinion* pairs and *Sent-Sent* pairs.

Sent-Opinion Pairs: RATION uses M_{align} to estimate alignment between review sentences and representative opinions (§4.1). To enable this learning, we construct *alignment* samples for fine-tuning M_{align} by pairing a sentence, s, from the ABSA dataset (X) with the representative opinion extracted from itself (Y) using the method described in §4.1. For *neutral* pairs, the second sentence, Y, is a representative opinion obtained from other sentences that have the same sentiment as s but discuss a different category. For *opposite* pairs, the second sentence, Y, is a representative opinion obtained from other sentences with the same category as s but an opposite sentiment.

Sent-Sent Pairs: RATION also uses M_{align} to estimate alignment between review sentences (§4.2). To enable this learning, we construct *alignment* samples as before for neutral pairs and opposite pairs except that instead of pairing sentences (X) with representative opinions extracted from randomly sampled sentences, we pair them with the sampled sentences themselves (Y). For alignment pairs, the second sentence Y are a randomly sampled sentence with the same aspect and sentiment as X.

5 Empirical Evaluation

We now describe experiments to evaluate RATION .

5.1 Implementation Detail

For the Opinion Extractor, RATION uses SemAE (Basu Roy Chowdhury et al., 2022) as the extractive summarization model but our method is independent of this choice. We only assume the existence of extractive summaries. We also perform experiments on the extractive summaries generated by Hercules (Hosking et al., 2023) (Appendix A.8). From the summarizing review sentences, RATION uses Snippext (Miao et al., 2020) as the ABSA model to extract representative opinions.

For the Rationales Extractor, to accelerate the calculation of the alignment probability, p_{align}, we use a sentiment classification model (Barbieri et al., 2020). Specifically, when the two input sentences do not have the same sentiment label, we directly set their p_{align} to 0. When extracting rationales, we extract clauses instead of full sentences since we find clauses are more specific to representative opinions than full sentences. We describe the process of dividing sentences into clauses in the appendix A.1. We also filter out rationale candidate set C with less than five sentences. When estimating popularity $pop(s)$, we use the default TextRank for an undirected graph to estimate the centrality of the node. For estimating $spec(s)$, we finetune a DeBERTa-base (He et al., 2020) model on the specificity dataset for 3 epochs with the learning rate as 2e-5 and batch size as 32. The weight of the diversity term γ is 0.1. As for Gibbs Sampling, η is 100 and θ is 200. When sampling from the conditional probability, we set the temperature of Softmax as 0.01.

For the alignment model M_{align}, we use one alignment models for the Space data and the Yelp dataset respectively. We first perform domain adaptation using sentences sampled from the corresponding train sets on RoBERTa-large (Liu et al., 2019) following steps described in Bar-Haim et al. (2021). To generate in-domain pairs to finetune the alignment model M_{align}, aside from sentences in the corresponding ABSA dataset, we additionally sample sentences from the corresponding train set to create a dataset containing 7,000 sentences. The annotations of the sampled sentences are predicted by the same ABSA model that RATION uses for the Opinion Extractor. For each sentence, we generate one Sent-Opinion pairs and Sent-Sent pairs for each label. We then perform down-sampling to create a dataset containing 24K samples for the Space data and the Yelp dataset respectively and use about 20K of them for training. We use the remaining samples for validation. The size of the dataset matches the size of ArgKP dataset (Bar-Haim et al., 2020) for the fair comparison we described in §5.5. We then finetune M_{align} on the in-domain datasets for 3 epochs with the learning rate as 1e-5 and batch size as 32.

5.2 Dataset

We perform the experiments on the Space dataset (Angelidis et al., 2021b) and the Yelp dataset[1]. For the Space dataset, we held out randomly sampled 250 entities with 100 reviews each as the test set. The remaining data was used for training and development. For the Yelp dataset, we perform cleaning and downsampling (Appendix A.2) and only retain entities whose categories contain 'restaurant'. From these entities, we sample 50 and 250 entities with 100 reviews each as the development set and the test set respectively. The statistics of datasets are shown in Appendix Table 5. We tune the hyperparameters on the development sets and report the performance on the test sets.

To finetune the ABSA model used in Opinion Extractor and produce fine-tuning samples for M_{align}, we use the ABSA dataset in the hotel domain for the Space dataset, and ACOS-restaurant dataset (Cai et al., 2021) for the Yelp dataset.

5.3 Rationale-based Summary Evaluation

We compare rationale-based opinion summaries generated by RATION with conventional summaries generated by a state-of-the-art opinion summarization model, SemAE (Basu Roy Chowdhury et al., 2022) using human evaluation. We ask annotators to compare the two types of summaries in a pairwise manner based on four criteria: which summary includes more information (*informativeness*), which summary contains less repeated phrases (*non-redundancy*), which summary is easier to read (*coherence*), and which summary is more useful for decision making (*usefulness*). We randomly sample 25 entities each from the test sets of the Space dataset and the Yelp dataset and generate 100-word summaries for each entity using RATION and SemAE. Each pair of summaries is annotated by three annotators recruited from Amazon Mechanical Turk (AMT). The human annotators are required to be in the United States, have HIT Approval Rate greater than 98, and be AMT masters. Fig. 8 in the Appendix shows a screenshot of our setup. We report the Best-worst scaling scores (Louviere et al., 2015) of RATION in Table 1.

	Info.	Non-Redun.	Cohe.	Use.
Space	20	**84**	**72**	**40**
Yelp	0	**100**	**64**	**32**

Table 1: Human comparison of rationale-based opinion summaries generated by RATION with conventional summaries generated by SemAE. **Bold** fonts indicate significant dDifferences (p<0.05, paired bootstrap resampling (Koehn, 2004)). Rationale-based opinion summaries outperform conventional opinion summaries on non-redundancy, coherence, and usefulness.

From the table we can see that, rationale-based summaries perform significantly better on non-redundancy, coherence, and usefulness than conventional summaries. Rationale-based summaries do not perform very well on informativeness because they pair each representative opinion with rationales. Therefore, while they provide more information per representative opinion, they understandably do not cover all opinions expressed in the conventional summaries because of the length limit. This can easily be fixed by increasing the length limit. We also perform error analysis of the summaries generated RATION (Appendix A.9). Overall, the experiments indicate that rationale-based summaries are less redundant, easier to read, and more useful for decision-making.

5.4 Rationale Evaluation

We evaluate the extracted rationales using automatic (§5.4.1) and human measures (§5.4.2).

5.4.1 Automatic Evaluation

We use the following four automatic measures for evaluating rationales for a given opinion.

To measure **relatedness** between the rationales and the corresponding representative opinion, (emb_{rel}), we use the average cosine similarity between the sentence embeddings (obtained using

[1] https://www.yelp.com/dataset

SimCSE (Gao et al., 2021)) of the representative opinion and each of its rationales.

To measure **specificity**, (key_{spec}), we use TF-IDF-based keywords. For this, we concatenate all review sentences belonging to the same rationale candidate set and calculate TF-IDF scores based on the concatenated sentences from each rationale candidate set of an entity. For each rationale candidate set, we extract five words with the highest TF-IDF scores that are not part of the representative opinions as the keywords. These keywords represent the popular details about the representative opinion but are not directly present in it. Given a set of rationales, key_{spec} is the sum of TF-IDF scores of the keywords covered by that set divided by the sum of TF-IDF scores of all keywords.

To measure **popularity**, (key_{pop}), we consider the fraction of rationales' tokens that are keywords. Given a set of rationales, key_{pop} is the sum of TF-IDF scores of the keywords covered by them divided by the sum of TF-IDF scores of all tokens present in the rationales.

To measure **diversity** among the rationales, (emb_{div}), we use one minus the average pairwise cosine similarity of their sentence embeddings.

Based on these four measures, we compare RATION with its variants: RATION (w/o X). RATION (w/o X) represents a variant of RATION that does not consider X for the probability of being rationales (Eqn. 3). We also compare RATION with InstructGPT (Ouyang et al., 2022) version 'gpt-3.5-turbo-0613'. To extract rationales using InstructGPT, the input is the representative opinion and the corresponding rationale candidate set and the instructions describe the four desirable properties of rationales (shown in Appendix A.4). We evaluate in two different settings: k=1 and k=3, where k is the number of rationales extracted for each representative opinion. The results are shown in Table 2. In addition to the four measures, we also report an *Overall* score which is the average of normalized values ($[0, 1]$) of these measures.

From the table, we can observe that in general, rationales generated by RATION outperform rationales generated by its variants considering the overall quality. This indicates that all terms in the probablity function (Eqn. 3) are important for extracting good rationales. For InstructGPT, we observe that although the instructions ask it to extract rationales with lots of details, some of the extracted rationales are paraphrases of the representative opinions, which is indicated by high emb_{rel} but poor key_{spec}. We provide more discussion of rationales generated by InstructGPT in Appendix A.5.

	emb_{rel}	key_{spec}	key_{pop}	emb_{div}	Overall
			Space (k=1)		
RATION	0.422	0.217	0.224	-	0.720
w/o *rel*	0.423	0.223	0.225	-	**0.750**
w/o *spec*	0.555	0.147	0.208	-	0.592
w/o *pop*	0.369	0.195	0.193	-	0.445
InstructGPT	0.586	0.139	0.129	-	0.333
			Space (k=3)		
RATION	0.414	0.498	0.224	0.580	**0.680**
w/o *rel*	0.415	0.499	0.223	0.575	0.668
w/o *spec*	0.508	0.420	0.222	0.528	0.471
w/o *pop*	0.377	0.465	0.203	0.627	0.468
w/o *div*	0.418	0.487	0.226	0.564	0.631
InstructGPT	0.501	0.438	0.193	0.530	0.300
			Yelp (k=1)		
RATION	0.358	0.202	0.233	-	**0.718**
w/o *rel*	0.372	0.201	0.226	-	0.713
w/o *spec*	0.469	0.154	0.208	-	0.626
w/o *pop*	0.313	0.180	0.198	-	0.504
InstructGPT	0.604	0.095	0.111	-	0.333
			Yelp (k=3)		
RATION	0.337	0.473	0.227	0.619	**0.649**
w/o *rel*	0.347	0.472	0.225	0.601	0.545
w/o *spec*	0.408	0.435	0.226	0.609	0.608
w/o *pop*	0.323	0.449	0.203	0.641	0.489
w/o *div*	0.345	0.455	0.228	0.596	0.486
InstructGPT	0.444	0.396	0.194	0.597	0.256

Table 2: Automatic evaluation of rationales on Space and Yelp datasets with one (k=1) and three (k=3) rationales extracted per representative opinion. Considering the four measures and their *overall* values, RATION extracts the best rationales.

5.4.2 Human Evaluation

We also conduct a human evaluation of the rationales generated by RATION and InstrutGPT (Ouyang et al., 2022) for a given representative opinion. We randomly sample 50 representative opinions each from the entities belonging to the test sets of the Space dataset and the Yelp dataset and generate three rationales for each representative opinion using RATION and InstructGPT. Each pair of rationale sets is evaluated by three annotators recruited from Amazon Mechanical Turk. The annotator details are same as in Sec. 5.3. We ask annotators to compare the two rationale sets in a pairwise manner based on three properties: relatedness, specificity, and diversity. Fig. 7 of the Appendix shows our setup. We report the Best-Worst Scaling scores of RATION in Table 3.

We can see from the table that RATION outperforms InstructGPT on specificity and diversity. For relatedness, both systems were judged to be comparable. Since most review sentences belonging to the rationale candidate set are already quite related

	Rel.	Spec.	Div.
Space	0	10	**20**
Yelp	-12	**48**	**54**

Table 3: Human evaluation of rationales generated by RATION and InstructGPT. **Bold** indicates significant differences (p<0.05, paired bootstrap resampling). RATION outperforms InstructGPT on specificity and diversity and is comparable to it on relatedness

.

	Silh	*NPMI*	*SC*	Overall
	Space			
RoBERTa$_{\text{mnli}}$	0.089	-0.061	0.970	0.779
KPA	0.134	-0.059	0.962	0.836
Snippext	0.108	-0.103	0.934	0.265
Hercules	0.009	-0.042	0.943	0.415
RATION	0.119	-0.051	0.969	**0.906**
	Yelp			
RoBERTa$_{\text{mnli}}$	0.015	-0.210	0.956	0.530
KPA	0.035	-0.208	0.934	0.758
Snippext	0.039	-0.265	0.805	0.318
RATION	0.040	-0.171	0.934	**0.953**

Table 4: Automatic evaluation of rationale candidate sets. Considering the three measures and their overall scores, RATION generates rationale candidates of better quality than the baselines.

to the representative opinion, it is understandable that both systems do not have much difference in relatedness. Overall, the experiments indicate that rationales extracted by RATION are more specific and diverse than InstructGPT.

5.5 Rationale Candidate Set Evaluation

RATION extracts rationales for a representative opinion from a rationale candidate set instead of all review sentences. In this experiment, we evaluate the goodness of this set by comparing it with adaptations of previous works that match a group of review sentences to summary sentences.

For this evaluation, we use three automatic measures. First, we view each rationale candidate set as a cluster of sentences and evaluate the clustering quality. We report Silhouettes scores (Rousseeuw, 1987) (*Silh*) based on the cosine similarity of the sentence embeddings. Second, we borrow measures from topic modeling to compute coherence of the sets using TF-IDF scores of tokens for coherence. Third, we also report the entailment score (*SC*) between the concatenation of all candidates in a rationale candidate set and the corresponding representative opinion as predicted by SummaC (Laban et al., 2022).

We compare RATION with four baseline models: RoBERTa$_{\text{mnli}}$, KPA, Snippext, and Hercules. RoBERTa$_{\text{mnli}}$ (Louis and Maynez, 2023) uses a RoBERTa-large (Liu et al., 2019) finetuned on the MNLI dataset (Williams et al., 2018) to match reviews to 'propositions'. KPA (Bar-Haim et al., 2021) uses a domain adapted RoBERTa-large that is then finetuned on ArgKP dataset (Bar-Haim et al., 2020) to match review sentences to 'key points'. Snippext (Miao et al., 2020) is trained on ABSA datasets to estimate the aspect and the sentiment distributions for each opinion and then uses similarity between these distributions to cluster 'opinions'. Hercules (Hosking et al., 2023) generates a path on a tree for each review sentence and summary and then uses path similarity to match review sentences to summary sentences. We use these models to estimate alignment between representative opinions and review sentences (details in Appendix A.7), and generate rationale candidate sets accordingly as in §4.2. Because of the different ranges of these measures, we normalize these measures to $[0, 1]$ among all baselines and use the average of these normalized metrics to evaluate the overall quality of rationale candidate sets. The results are shown in Table 4. In addition to the three measures, we also report an *Overall* score which is the average of normalized values ($[0, 1]$) of these measures.

From the table, we can observe that the rationale candidate sets generated by RATION have the best overall performance. The result shows the effectiveness of the in-domain pairs we created to finetune the alignment model.

6 Conclusion

We propose rationale-based opinion summarization, a new paradigm for summarizing reviews. The rationale-based summaries present representative opinions and their corresponding rationales. We define four desirable properties of rationales: relatedness, specificity, popularity, and diversity. Based on these properties, we propose RATION , an unsupervised extractive system that extracts representative opinions and their corresponding rationales based on Gibbs sampling. Our experiments show that rationale-based summaries generated by RATION are more useful than conventional opinion summaries. Our experiments also show that the rationales generated by RATION outperform its variants and strong baselines.

7 Limitation

Since there is no supervision for extracting rationales, RATION separately estimates the four properties of rationales separately and assign equal importance to relatedness, specificity, and popularity. Future work can collect supervised data to extract rationales and build a system that can jointly model and assign weights to the four properties based on the supervised data. Second limitation is during extracting rationales, RATION does not consider the similarity between representative opinions. Another limitation is that our datasets and all experiments are only focused on the English language.

8 Ethical Consideration

We do not expect any ethical risks caused by our work. The datasets we use are all publicly available. We do not annotate any data on our own. We performed human evaluation experiments on Amazon Mechanical Turk. The annotators were compensated at a rate of $15 per hour. During the evaluation, human annotators were not exposed to any sensitive or explicit content.

References

Reinald Kim Amplayo and Mirella Lapata. 2020. Unsupervised opinion summarization with noising and denoising. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1934–1945, Online. Association for Computational Linguistics.

Stefanos Angelidis, Reinald Kim Amplayo, Yoshihiko Suhara, Xiaolan Wang, and Mirella Lapata. 2021a. Extractive opinion summarization in quantized transformer spaces. Transactions of the Association for Computational Linguistics, 9:277–293.

Stefanos Angelidis, Reinald Kim Amplayo, Yoshihiko Suhara, Xiaolan Wang, and Mirella Lapata. 2021b. Extractive opinion summarization in quantized transformer spaces. Transactions of the Association for Computational Linguistics, 9:277–293.

Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. 2020. From arguments to key points: Towards automatic argument summarization. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4029–4039, Online. Association for Computational Linguistics.

Roy Bar-Haim, Lilach Eden, Yoav Kantor, Roni Friedman, and Noam Slonim. 2021. Every bite is an experience: Key Point Analysis of business reviews. 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), pages 3376–3386, Online. Association for Computational Linguistics.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1644–1650, Online. Association for Computational Linguistics.

Somnath Basu Roy Chowdhury, Chao Zhao, and Snigdha Chaturvedi. 2022. Unsupervised extractive opinion summarization using sparse coding. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1209–1225, Dublin, Ireland. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.".

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2019. Unsupervised opinion summarization as copycat-review generation. arXiv preprint arXiv:1911.02247.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. 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), pages 340–350, Online. Association for Computational Linguistics.

Cindy Man-Yee Cheung, Choon-Ling Sia, and Kevin KY Kuan. 2012. Is this review believable? a study of factors affecting the credibility of online consumer reviews from an elm perspective. Journal of the Association for Information Systems, 13(8):2.

Eric Chu and Peter Liu. 2019. Meansum: A neural model for unsupervised multi-document abstractive summarization. In International Conference on Machine Learning, pages 1223–1232. PMLR.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In Empirical Methods in Natural Language Processing (EMNLP).

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In International Conference on Learning Representations.

Tom Hosking, Hao Tang, and Mirella Lapata. 2023. Attributable and scalable opinion summarization. In Proceedings of the 61st Annual Meeting of the

Association for Computational Linguistics (Volume 1: Long Papers), pages 8488–8505, Toronto, Canada. Association for Computational Linguistics.

Hayate Iso, Xiaolan Wang, Yoshihiko Suhara, Stefanos Angelidis, and Wang-Chiew Tan. 2021. Convex aggregation for opinion summarization. In Findings of the Association for Computational Linguistics: EMNLP 2021, pages 3885–3903.

Masaru Isonuma, Junichiro Mori, Danushka Bollegala, and Ichiro Sakata. 2021. Unsupervised abstractive opinion summarization by generating sentences with tree-structured topic guidance. Transactions of the Association for Computational Linguistics, 9:945–961.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Wei-Jen Ko, Greg Durrett, and Junyi Jessy Li. 2019. Domain agnostic real-valued specificity prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 6610–6617.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. Transactions of the Association for Computational Linguistics, 10:163–177.

Haoyuan Li, Somnath Basu Roy Chowdhury, and Snigdha Chaturvedi. 2023. Aspect-aware unsupervised extractive opinion summarization. In Findings of the Association for Computational Linguistics: ACL 2023, pages 12662–12678.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Annie Louis and Joshua Maynez. 2023. OpineSum: Entailment-based self-training for abstractive opinion summarization. In Findings of the Association for Computational Linguistics: ACL 2023, pages 10774–10790, Toronto, Canada. Association for Computational Linguistics.

Jordan J Louviere, Terry N Flynn, and Anthony Alfred John Marley. 2015. Best-worst scaling: Theory, methods and applications. Cambridge University Press.

Zhengjie Miao, Yuliang Li, Xiaolan Wang, and Wang-Chiew Tan. 2020. Snippext: Semi-supervised opinion mining with augmented data. In Proceedings of The Web Conference 2020, pages 617–628.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35:27730–27744.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NL pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20:53–65.

Yoshihiko Suhara, Xiaolan Wang, Stefanos Angelidis, and Wang-Chiew Tan. 2020. Opiniondigest: A simple framework for opinion summarization. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5789–5798.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1112–1122. Association for Computational Linguistics.

A Appendix

A.1 Text Segmentation

Due to the nature of reviews, many review sentences discuss several unrelated aspects, such as ‘The room is spacious and staff are helpful.’ . These sentences might make rationales less specific to the representative opinions because they might contain unrelated information concerning a certain opinion. To alleviate these problems, RATION extracts clauses from review sentences using a constituency parser (Kitaev and Klein, 2018). The goal of the extraction is to reach a balance of two criteria. First, the resulting clauses are complete and fluent sentences. Second, the most resulting clauses only discuss one aspect.

Given a parse tree of a sentence, RATION traverse it from its root to determine the boundary of a clause. When a node whose tag is ‘S’ is traversed, if it has not been extracted yet, RATION will check the length of the corresponding clause. For clauses longer than the maximum length ϵ, they still might discuss several aspects. Therefore, RATION further traverse all their children as in Fig. 3c. For clauses shorter than the minimum length γ, the clauses might be incomplete and the traversal stops at these nodes. The traversal also stops at the node whose tag is ‘SBAR’ since the corresponding clauses usually complement other clauses. If the length of the corresponding clause is between the maximum length ϵ and γ, RATION will extract the clause as in Fig. 3a. If RATION only extracts one clause from the sentence, RATION will extract the whole sentence instead to keep the information complete as in Fig. 3b. If RATION extracts more than one clause from the sentence, RATION will further check the distances between neighboring clauses. If the distance between any two neighboring clauses is larger than γ, RATION will also extract the whole sentence. Otherwise, RATION extracts the clauses. The above process extracts as many clauses as possible while keeping the extracting clauses complete. In the experiment, we set the maximum length as 20 and minimum length as 2.

A.2 Preprocessing of Yelp Dataset

For yelp dataset, we remove entities that contain less than 20 reviews. For entities containing more than 200 reviews, we randomly sample 200 reviews and discard other reviews of the entities to prevent dominant influences of some entities. For the remaining entities, we perform downsampling to create a dataset containing around 14.9K entities and around 1.22M reviews. The statistics are show in Table 5.

Dataset	Train	Dev.	Test
Space	11.2K/1.10M	50/5K	250/25K
Yelp	14.9K/1.22M	50/5K	250/25K

Table 5: Dataset statistics for Space and Yelp. We report entity/review for each split of two datasets.

Opinion Cluster	Keyword
location is great	seattle downtown vintage library walk
bed is super comfortable bed is great	pillow linen comfy ever mattress

Table 6: Samples of opinion clusters and keywords extracted from their rationale candidates. Keywords are shown in descending order of TF-IDF. Most keywords represent details highly related to but not repetitive of the corresponding opinion groups.

A.3 Preprocessing for Keyword Extraction

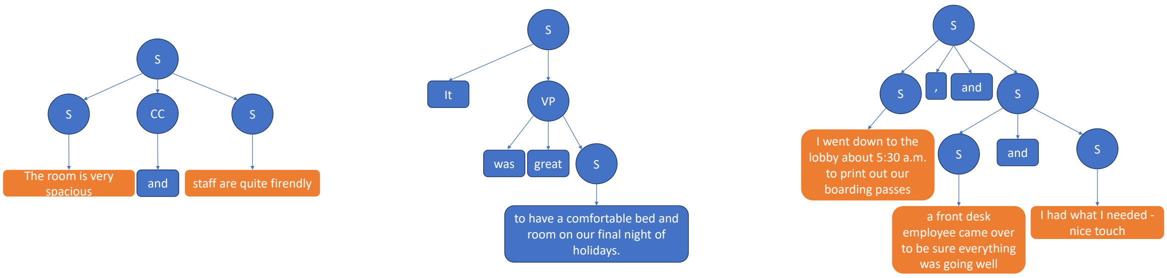
In §5.4 and §5.5, we extract keywords to evaluate the performance of RATION . For this purpose, we perform the standard preprocessing. We first remove stop words using NLTK (Bird et al., 2009) and filter out extreme words using Gensim (Řehůřek and Sojka, 2010). We finally perform lemmatization using NLTK. We show examples of extracted keywords in Table 6.

A.4 Instruction for InstructGPT

To extract rationales using InstructGPT, we provide the instructions that describe the four desirable properties of rationales as well as the representative opinion and its corresponding rationales. Under the extractive setting, we try several variations of prompts including paraphrasing, reordering, and restructuring the instruction material. We show the best instruction that we use for extracting one rationales for each opinion in Figure 5 and extracting three rationale for each opinion in Figure 6.

A.5 Error Analysis of InstructGPT Rationale

As discussed in Section 5.4.1, the performance of InstructGPT was encouraging for an initial study but not up to the mark. Specifically, we manually analyzed the InstructGPT’s rationales while also asking for explanations of those rationales. We found that the rationales extracted by the InstructGPT were lacking in many senses. First, the In-

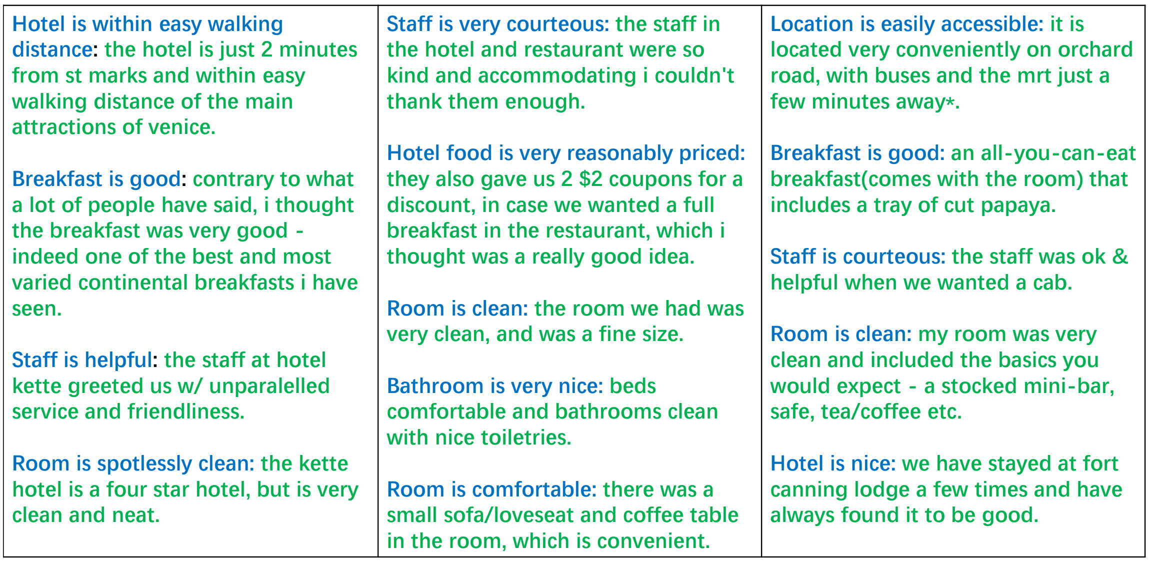

S
S
CC
S
The room is very spacious
and
staff are quite firendly
S
It
VP
was
great
S
to have a comfortable bed and room on our final night of holidays.
S
S
,
and
S
I went down to the lobby about 5:30 a.m. to print out our boarding passes
S
and
S
a front desk employee came over to be sure everything was going well
I had what I needed - nice touch

(a) The root has two clause children and therefore two corresponding clauses are extracted from the sentence.

(b) The whole sentence is extracted to keep the information complete since there is only one clause in the sentence.

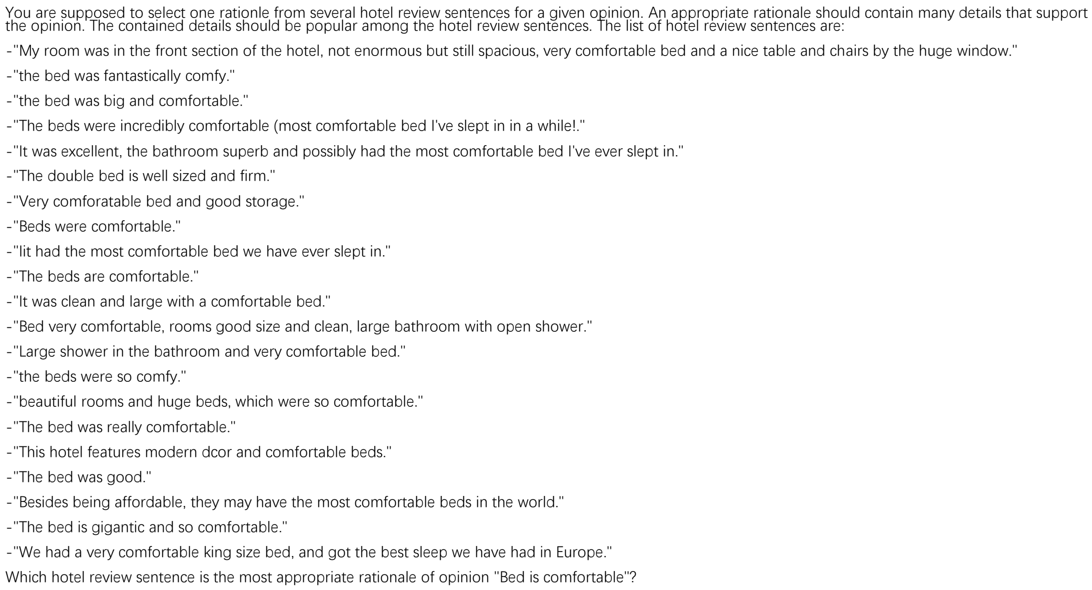
(c) The root has two children clause. Since the length of the second clause is longer than the maximum length, `RATION` traverse its children and find two children clause. Therefore, three clause are extracted from the sentence.

Figure 3: Three sentences and their constituency parsing trees. A orange box denotes one extracted clause.



Hotel is within easy walking distance: the hotel is just 2 minutes from st marks and within easy walking distance of the main attractions of venice. Breakfast is good: contrary to what a lot of people have said, i thought the breakfast was very good - indeed one of the best and most varied continental breakfasts i have seen. Staff is helpful: the staff at hotel kette greeted us w/ unparalelled service and friendliness. Room is spotlessly clean: the kette hotel is a four star hotel, but is very clean and neat.	Staff is very courteous: the staff in the hotel and restaurant were so kind and accommodating i couldn't thank them enough. Hotel food is very reasonably priced: they also gave us 2 $2 coupons for a discount, in case we wanted a full breakfast in the restaurant, which i thought was a really good idea. Room is clean: the room we had was very clean, and was a fine size. Bathroom is very nice: beds comfortable and bathrooms clean with nice toiletries. Room is comfortable: there was a small sofa/loveseat and coffee table in the room, which is convenient.	Location is easily accessible: it is located very conveniently on orchard road, with buses and the mrt just a few minutes away*. Breakfast is good: an all-you-can-eat breakfast(comes with the room) that includes a tray of cut papaya. Staff is courteous: the staff was ok & helpful when we wanted a cab. Room is clean: my room was very clean and included the basics you would expect - a stocked mini-bar, safe, tea/coffee etc. Hotel is nice: we have stayed at fort canning lodge a few times and have always found it to be good.

Figure 4: Three sample rationale-based summaries. Each line presents a representative opinion and its rationale.



You are supposed to select one rationle from several hotel review sentences for a given opinion. An appropriate rationale should contain many details that support the opinion. The contained details should be popular among the hotel review sentences. The list of hotel review sentences are:

-"My room was in the front section of the hotel, not enormous but still spacious, very comfortable bed and a nice table and chairs by the huge window."

-"the bed was fantastically comfy."

-"the bed was big and comfortable."

-"The beds were incredibly comfortable (most comfortable bed I've slept in in a while!."

-"It was excellent, the bathroom superb and possibly had the most comfortable bed I've ever slept in."

-"The double bed is well sized and firm."

-"Very comforatable bed and good storage."

-"Beds were comfortable."

-"Iit had the most comfortable bed we have ever slept in."

-"The beds are comfortable."

-"It was clean and large with a comfortable bed."

-"Bed very comfortable, rooms good size and clean, large bathroom with open shower."

-"Large shower in the bathroom and very comfortable bed."

-"the beds were so comfy."

-"beautiful rooms and huge beds, which were so comfortable."

-"The bed was really comfortable."

-"This hotel features modern dcor and comfortable beds."

-"The bed was good."

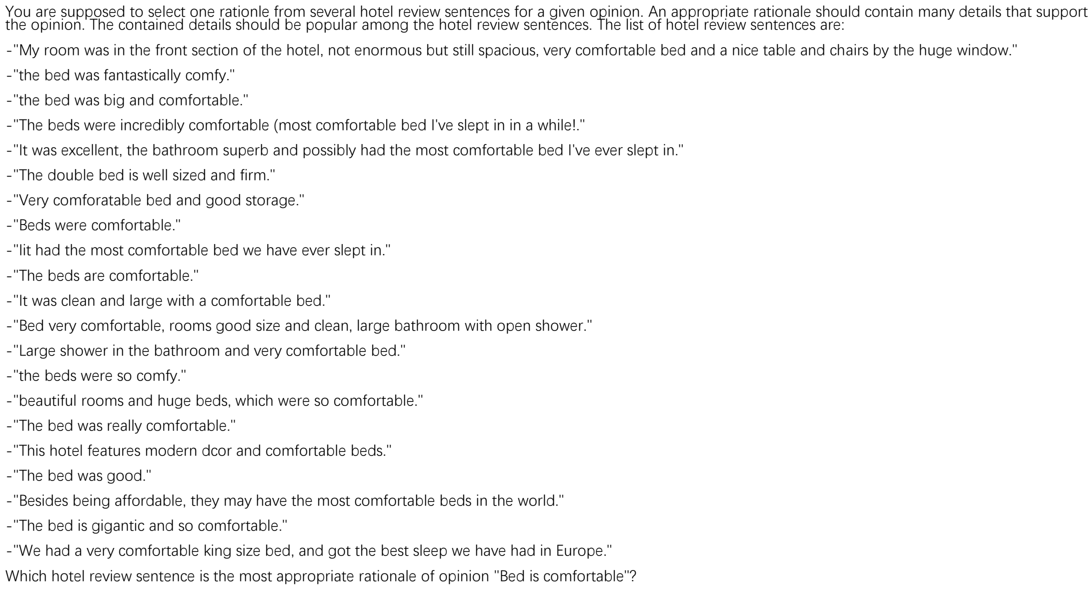
-"Besides being affordable, they may have the most comfortable beds in the world."

-"The bed is gigantic and so comfortable."

-"We had a very comfortable king size bed, and got the best sleep we have had in Europe."

Which hotel review sentence is the most appropriate rationale of opinion "Bed is comfortable"?

Figure 5: Example instruction for extracting one rationale for each representative opinion using InstructGPT.



You are supposed to select one rationle from several hotel review sentences for a given opinion. An appropriate rationale should contain many details that support the opinion. The contained details should be popular among the hotel review sentences. The list of hotel review sentences are:

-"My room was in the front section of the hotel, not enormous but still spacious, very comfortable bed and a nice table and chairs by the huge window."

-"the bed was fantastically comfy."

-"the bed was big and comfortable."

-"The beds were incredibly comfortable (most comfortable bed I've slept in in a while!."

-"It was excellent, the bathroom superb and possibly had the most comfortable bed I've ever slept in."

-"The double bed is well sized and firm."

-"Very comforatable bed and good storage."

-"Beds were comfortable."

-"Iit had the most comfortable bed we have ever slept in."

-"The beds are comfortable."

-"It was clean and large with a comfortable bed."

-"Bed very comfortable, rooms good size and clean, large bathroom with open shower."

-"Large shower in the bathroom and very comfortable bed."

-"the beds were so comfy."

-"beautiful rooms and huge beds, which were so comfortable."

-"The bed was really comfortable."

-"This hotel features modern dcor and comfortable beds."

-"The bed was good."

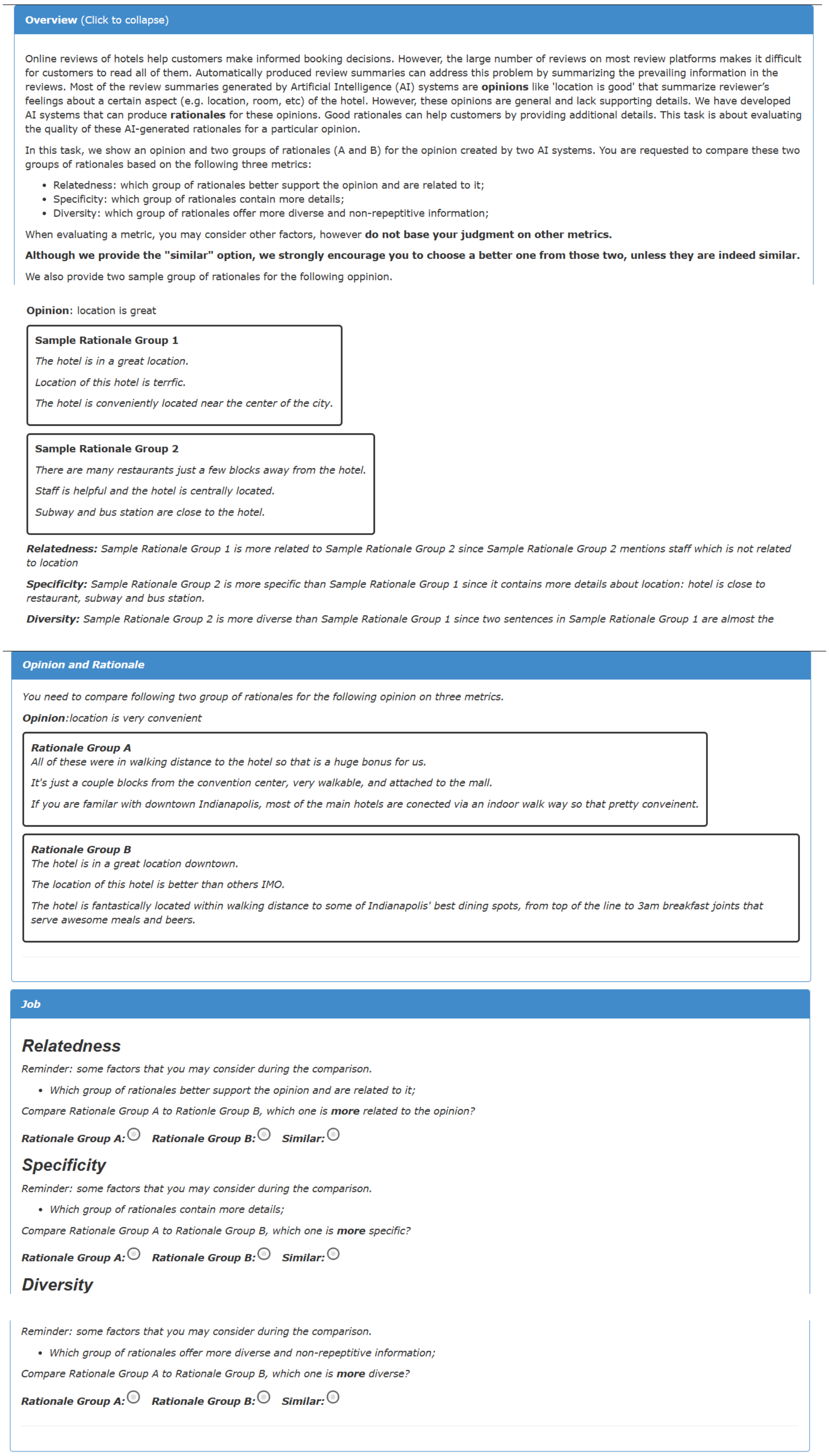
-"Besides being affordable, they may have the most comfortable beds in the world."

-"The bed is gigantic and so comfortable."

-"We had a very comfortable king size bed, and got the best sleep we have had in Europe."

Which hotel review sentence is the most appropriate rationale of opinion "Bed is comfortable"?

Figure 6: Example instruction for extracting three rationales for each representative opinion using InstructGPT.



Overview (Click to collapse)

Online reviews of hotels help customers make informed booking decisions. However, the large number of reviews on most review platforms makes it difficult for customers to read all of them. Automatically produced review summaries can address this problem by summarizing the prevailing information in the reviews. Most of the review summaries generated by Artificial Intelligence (AI) systems are **opinions** like 'location is good' that summarize reviewer's feelings about a certain aspect (e.g. location, room, etc) of the hotel. However, these opinions are general and lack supporting details. We have developed AI systems that can produce **rationales** for these opinions. Good rationales can help customers by providing additional details. This task is about evaluating the quality of these AI-generated rationales for a particular opinion.

In this task, we show an opinion and two groups of rationales (A and B) for the opinion created by two AI systems. You are requested to compare these two groups of rationales based on the following three metrics:

- Relatedness: which group of rationales better support the opinion and are related to it;
- Specificity: which group of rationales contain more details;
- Diversity: which group of rationales offer more diverse and non-repeptitive information;

When evaluating a metric, you may consider other factors, however **do not base your judgment on other metrics.**

Although we provide the "similar" option, we strongly encourage you to choose a better one from those two, unless they are indeed similar.

We also provide two sample group of rationales for the following oppinion.

Opinion: location is great

Sample Rationale Group 1

The hotel is in a great location.

Location of this hotel is terrfic.

The hotel is conveniently located near the center of the city.

Sample Rationale Group 2

There are many restaurants just a few blocks away from the hotel.

Staff is helpful and the hotel is centrally located.

Subway and bus station are close to the hotel.

Relatedness: *Sample Rationale Group 1 is more related to Sample Rationale Group 2 since Sample Rationale Group 2 mentions staff which is not related to location*

Specificity: *Sample Rationale Group 2 is more specific than Sample Rationale Group 1 since it contains more details about location: hotel is close to restaurant, subway and bus station.*

Diversity: *Sample Rationale Group 2 is more diverse than Sample Rationale Group 1 since two sentences in Sample Rationale Group 1 are almost the*

Opinion and Rationale

You need to compare following two group of rationales for the following opinion on three metrics.

Opinion*:location is very convenient*

Rationale Group A
All of these were in walking distance to the hotel so that is a huge bonus for us.

It's just a couple blocks from the convention center, very walkable, and attached to the mall.

If you are familar with downtown Indianapolis, most of the main hotels are conected via an indoor walk way so that pretty conveinent.

Rationale Group B
The hotel is in a great location downtown.

The location of this hotel is better than others IMO.

The hotel is fantastically located within walking distance to some of Indianapolis' best dining spots, from top of the line to 3am breakfast joints that serve awesome meals and beers.

Job

Relatedness

Reminder: some factors that you may consider during the comparison.

- *Which group of rationales better support the opinion and are related to it;*

*Compare Rationale Group A to Rationle Group B, which one is **more** related to the opinion?*

Rationale Group A: ○ ***Rationale Group B:*** ○ ***Similar:*** ○

Specificity

Reminder: some factors that you may consider during the comparison.

- *Which group of rationales contain more details;*

*Compare Rationale Group A to Rationale Group B, which one is **more** specific?*

Rationale Group A: ○ ***Rationale Group B:*** ○ ***Similar:*** ○

Diversity

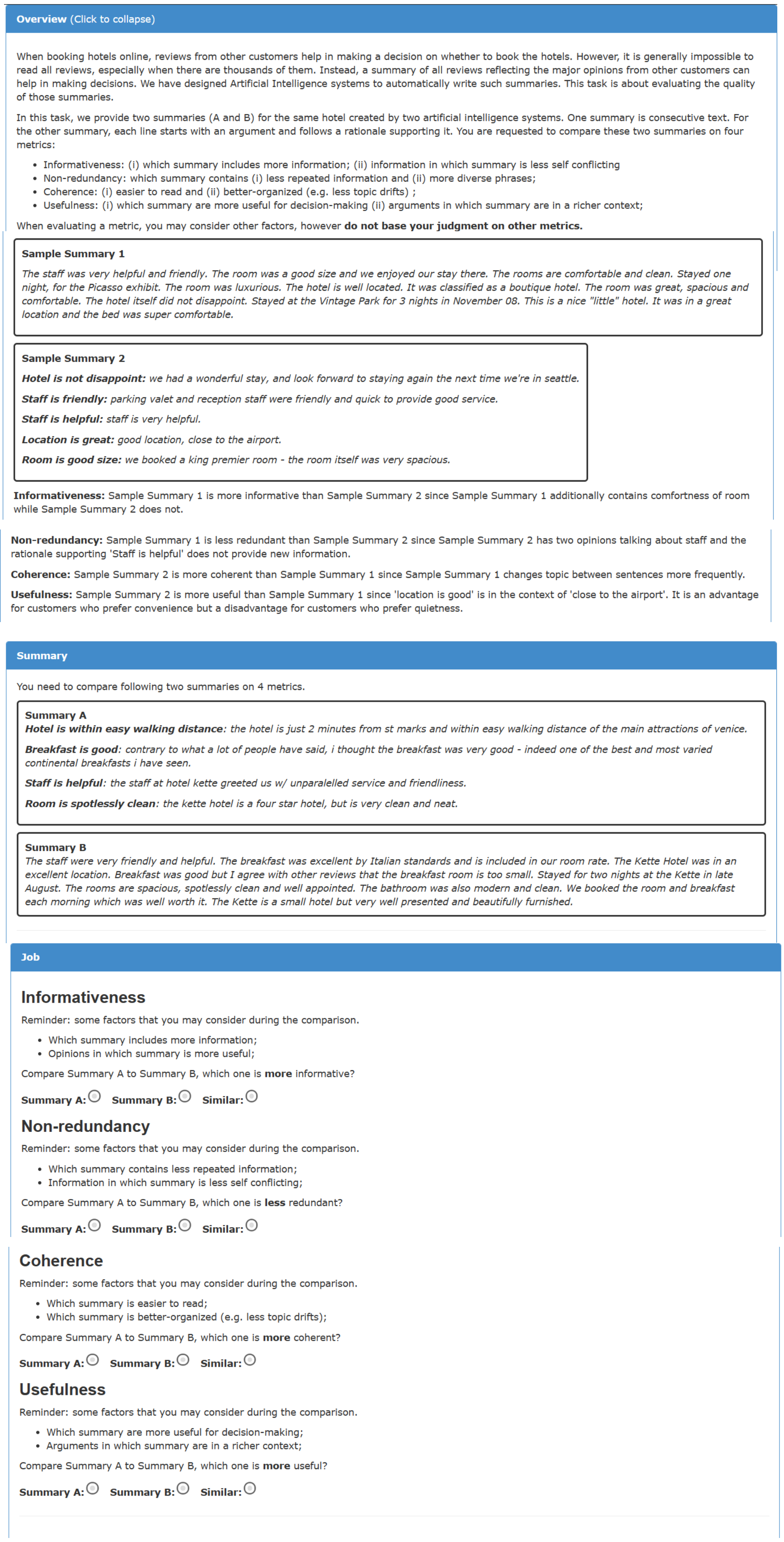
Reminder: some factors that you may consider during the comparison.

- *Which group of rationales offer more diverse and non-repeptitive information;*

*Compare Rationale Group A to Rationale Group B, which one is **more** diverse?*

Rationale Group A: ○ ***Rationale Group B:*** ○ ***Similar:*** ○

Figure 7: AMT instructions for human evaluation for comparing rationales.



Overview (Click to collapse)

When booking hotels online, reviews from other customers help in making a decision on whether to book the hotels. However, it is generally impossible to read all reviews, especially when there are thousands of them. Instead, a summary of all reviews reflecting the major opinions from other customers can help in making decisions. We have designed Artificial Intelligence systems to automatically write such summaries. This task is about evaluating the quality of those summaries.

In this task, we provide two summaries (A and B) for the same hotel created by two artificial intelligence systems. One summary is consecutive text. For the other summary, each line starts with an argument and follows a rationale supporting it. You are requested to compare these two summaries on four metrics:

- Informativeness: (i) which summary includes more information; (ii) information in which summary is less self conflicting
- Non-redundancy: which summary contains (i) less repeated information and (ii) more diverse phrases;
- Coherence: (i) easier to read and (ii) better-organized (e.g. less topic drifts) ;
- Usefulness: (i) which summary are more useful for decision-making (ii) arguments in which summary are in a richer context;

When evaluating a metric, you may consider other factors, however **do not base your judgment on other metrics.**

Sample Summary 1

The staff was very helpful and friendly. The room was a good size and we enjoyed our stay there. The rooms are comfortable and clean. Stayed one night, for the Picasso exhibit. The room was luxurious. The hotel is well located. It was classified as a boutique hotel. The room was great, spacious and comfortable. The hotel itself did not disappoint. Stayed at the Vintage Park for 3 nights in November 08. This is a nice "little" hotel. It was in a great location and the bed was super comfortable.

Sample Summary 2

Hotel is not disappoint: *we had a wonderful stay, and look forward to staying again the next time we're in seattle.*

Staff is friendly: *parking valet and reception staff were friendly and quick to provide good service.*

Staff is helpful: *staff is very helpful.*

Location is great: *good location, close to the airport.*

Room is good size: *we booked a king premier room - the room itself was very spacious.*

Informativeness: Sample Summary 1 is more informative than Sample Summary 2 since Sample Summary 1 additionally contains comfortness of room while Sample Summary 2 does not.

Non-redundancy: Sample Summary 1 is less redundant than Sample Summary 2 since Sample Summary 2 has two opinions talking about staff and the rationale supporting 'Staff is helpful' does not provide new information.

Coherence: Sample Summary 2 is more coherent than Sample Summary 1 since Sample Summary 1 changes topic between sentences more frequently.

Usefulness: Sample Summary 2 is more useful than Sample Summary 1 since 'location is good' is in the context of 'close to the airport'. It is an advantage for customers who prefer convenience but a disadvantage for customers who prefer quietness.

Summary

You need to compare following two summaries on 4 metrics.

Summary A
Hotel is within easy walking distance: *the hotel is just 2 minutes from st marks and within easy walking distance of the main attractions of venice.*

Breakfast is good*: contrary to what a lot of people have said, i thought the breakfast was very good - indeed one of the best and most varied continental breakfasts i have seen.*

Staff is helpful*: the staff at hotel kette greeted us w/ unparalelled service and friendliness.*

Room is spotlessly clean*: the kette hotel is a four star hotel, but is very clean and neat.*

Summary B
The staff were very friendly and helpful. The breakfast was excellent by Italian standards and is included in our room rate. The Kette Hotel was in an excellent location. Breakfast was good but I agree with other reviews that the breakfast room is too small. Stayed for two nights at the Kette in late August. The rooms are spacious, spotlessly clean and well appointed. The bathroom was also modern and clean. We booked the room and breakfast each morning which was well worth it. The Kette is a small hotel but very well presented and beautifully furnished.

Job

Informativeness

Reminder: some factors that you may consider during the comparison.

- Which summary includes more information;
- Opinions in which summary is more useful;

Compare Summary A to Summary B, which one is **more** informative?

Summary A: ○ **Summary B:** ○ **Similar:** ○

Non-redundancy

Reminder: some factors that you may consider during the comparison.

- Which summary contains less repeated information;
- Information in which summary is less self conflicting;

Compare Summary A to Summary B, which one is **less** redundant?

Summary A: ○ **Summary B:** ○ **Similar:** ○

Coherence

Reminder: some factors that you may consider during the comparison.

- Which summary is easier to read;
- Which summary is better-organized (e.g. less topic drifts);

Compare Summary A to Summary B, which one is **more** coherent?

Summary A: ○ **Summary B:** ○ **Similar:** ○

Usefulness

Reminder: some factors that you may consider during the comparison.

- Which summary are more useful for decision-making;
- Arguments in which summary are in a richer context;

Compare Summary A to Summary B, which one is **more** useful?

Summary A: ○ **Summary B:** ○ **Similar:** ○

Figure 8: AMT instructions for human evaluation of comparing summaries.

structGPT might ignore the part of the instruction that required the rationales to have additional details as compared to the opinions. The extracted rationales were simply paraphrases of the opinions. This defeats the purpose of having rationales. Second, the InstructGPT might misunderstand what is meant by "containing additional details". For example, it might focus too much on plural forms or tenses of certain words. The InstructGPT might think "Rooms are great" is an appropriate rationale for "Room is great" because maybe "Rooms are great" suggests there are many rooms that are great instead of one room. We faced these problems even after trying multiple prompts. In the future, as LLMs hopefully improve, future works could revisit this problem for better solutions.

A.6 Human Evaluation

The human annotators are required to be in the United States, have HIT Approval Rate greater than 98, and be masters. The screenshot of the human evaluation interface for rationale evaluation is shown in Figure 7. The screenshot of the human evaluation interface for summary evaluation is shown in Figure 8.

A.7 Implementation Detail of Rationale Candidate Set Evaluation

We compare RATION with four baseline models: RoBERTa$_{\text{mnli}}$, KPA, Snippext, and Hercules. RoBERTa$_{\text{mnli}}$ uses RoBERTa-large (Liu et al., 2019) finetuned on the MNLI dataset (Williams et al., 2018). RoBERTa$_{\text{mnli}}$ then uses the finetuned model to estimate the alignment probability between review sentences and representative opinions

KPA uses RoBERTa-large (Liu et al., 2019) as the base model and performs the same domain adaptation as RATION . The model is then finetuned on ArgKP dataset using the same hyperparameters as (Bar-Haim et al., 2021). KPA then uses the finetuned model to estimate the alignment probability.

Snippext uses the ABSA model (Miao et al., 2020). Snippext estimates the aspect distribution and the sentiment distribution for each representative opinion and review sentence based on the ABSA model. Snippext then estimates the alignment probability as the product of the cosine similarity of aspect distribution and the sentiment distribution between representaitive opinions and review sentences.

Hercules generates the path representation on a tree for each representative opinions and review sentences. If a representative opinion and a review sentence has the same first node of their path representation, the review sentence belongs to the rationale candidate set of that representative opinion.

For a fair comparison, we extract an average of 8 rationale candidate sets for each entity and all rationale candidate sets on average cover 30% of review sentences except for Hercules. When using SimCSE to obtain sentence representations, we use 'unsup-simcse-roberta-large' version.

	Silh	*NPMI*	*SC*	Overall
		Space		
RoBERTa$_{\text{mnli}}$	0.088	-0.034	0.953	0.693
KPA	0.142	-0.013	0.946	0.925
Snippext	0.122	-0.065	0.916	0.270
Hercules	0.009	-0.064	0.944	0.263
RATION	0.135	-0.012	0.952	**0.974**

Table 7: Automatic evaluation of rationale candidate sets when the representative opinions are extracted from summaries generated by Hercules. Considering the three measures and their overall scores, RATION still generates rationale candidates of better quality than the baselines when using the other extractive opinion summarization system.

A.8 Experiment with Hercules

RATION is independent of the choice of extractive summarization systems and can work with other extractive summarization systems. In this section, we show the automatic metrics on the Space dataset when the representative opinions are extracted from the summaries produced by the extractive version of Hercules. All the implementation details are the same.

We show the automatic metric for evaulating rationale candidate sets in Table 7. It can be observed that RATION still generates rationale candidates of better quality than the baselines when using the other extractive opinion summarization system, which also shows RATION is independent of extractive summarization systems.

We show the automatic metrics for evaulating rationales in Table 8. It can be observed that RATION also extracts the best rationales when the representative opinions are extracted from the summaries produced by Hercules.

	emb_{rel}	key_{spec}	key_{pop}	emb_{div}	Overall
			Space (k=1)		
RATION	0.399	0.236	0.237	-	0.728
w/o rel	0.400	0.239	0.237	-	**0.748**
w/o $spec$	0.525	0.174	0.219	-	0.554
w/o pop	0.349	0.212	0.207	-	0.389
InstructGPT	0.558	0.167	0.172	-	0.333
			Space (k=3)		
RATION	0.390	0.520	0.239	0.577	**0.623**
w/o rel	0.391	0.524	0.238	0.572	0.614
w/o $spec$	0.474	0.456	0.240	0.546	0.485
w/o pop	0.351	0.496	0.222	0.631	0.399
w/o div	0.398	0.511	0.241	0.555	0.575

Table 8: Automatic evaluation of rationales on the Space dataset with one (k=1) and three (k=3) rationales extracted per representative opinion. Considering the four measures and their *overall* values, RATION still extracts the best rationales when the representative opinions are extracted from summaries generated by the other extractive summarization system.

A.9 Error Analysis

RATION occasionally generates undesirable rationale-based opinion summaries. We analyze these summaries and find the most common errors are the extracted rationales of an opinion not containing many related details of that opinion. For example, in the right sample of Figure 4, the extracted rationale for the opinion ‘Room is clean’, ‘my room was very clean and included the basics you would expect - a stocked mini-bar, safe, tea/coffe etc.’, only mentions ‘clean’ and contains lots of details not related to the detail. The main reason is that RATION separately estimates the specificity and relatedness as mentioned in Section 7. Suppose a sentence discusses aspect X and aspect Y, and it only briefly mentions X but contains lots of details related to Y. When extracting rationales for an opinion about aspects X, the sentence would have a high relatedness score because it mentions X. It would also have a high specificity score because it contains many details. We reduce such errors by dividing review sentences into clauses and extracting clauses as rationales (Appendix A.1). However, some resulting clauses might still discuss multiple aspects. Future work can explore how to jointly model these four properties at the same time.

We also find other less frequent errors, such as some representative opinions being too similar and the alignment model making wrong estimations.