Aspect-based Key Point Analysis for Quantitative Summarization of Reviews

Anonymous ACL submission

Abstract

Key Point Analysis (KPA) is originally for summarizing arguments, where short sentences containing salient viewpoints are extracted as key points (KPs) and quantified for their prevalence as salience scores. Recently, KPA was applied to summarize reviews, but the study still relies on sentence-based KP extraction and matching, which leads to two issues: sentence-based extraction can result in KPs of overlapping opinions on the same aspects, and sentence-based matching of KP to review comment can be inaccurate, resulting in inaccurate salience scores. To address the above issues, in this paper, we propose Aspect-based Key Point Analysis (ABKPA), a novel framework for quantitative review summarization. Leveraging the readily available aspect-based sentiment analysis (ABSA) resources of reviews to automatically annotate silver labels for matching aspect-sentiment pairs, we propose a contrastive learning model to effectively match KPs to reviews and quantify KPs at the aspect level. Especially, the framework ensures extracting KP of distinct aspects and opinions, leading to more accurate opinion quantification. Experiments on five business categories of the popular Yelp review dataset show that ABKPA outperforms state-of-the-art baselines. Source code and data are available at: `https://anonymous.4open.science/r/ABKPA-A233`

1 Introduction

Summarization of user reviews on the online marketplace has become essential both for businesses to improve their product and service qualities and for customers to make purchasing decisions. Although the star ratings aggregated from customer reviews are widely used to measure quality of service for business entities (McGlohon et al., 2010; Tay et al., 2020), they can not explain specific details to achieve business intelligence and informed decisions. Early studies on review summarization focus on textual summaries that only represent the major opinions in reviews (Dash et al., 2019; Shandilya et al., 2018) but ignore the minority opinions and fail to quantify the opinion prevalence.

Recently, the quantitative view was introduced to review summarization under the novel framework named Key Point Analysis (KPA) (Bar-Haim et al., 2020a,b, 2021). KPA studies were initially extractive and developed for argument summarization (Bar-Haim et al., 2020a,b), and are then adapted for business reviews (Bar-Haim et al., 2021). KPA consists of two subtasks, namely Key Point extraction, which extracts salient sentences as KPs, and Key Point Matching, which quantifies the prevalence of KPs as the number of matching comments in reviews [1]. More recent KPA studies used abstractive summarization models to generate salient KPs (Kapadnis et al., 2021; Li et al., 2023a).

Whether extractive or abstractive approaches, existing KPA studies still perform KP extraction and matching at the sentence level, which has two major issues. First, the extracted KPs (i.e., short sentences) can contain redundant opinions on the same aspect. Subsequently, with both comments and KPs containing multiple opinions, sentence-based matching of KPs to comment then becomes ineffective and results in inaccurate quantification for KP prevalence.

To address the two above issues, we propose Aspect-based Key Point Analysis (ABKPA), a novel KPA framework for quantitative review summarization. ABKPA comprises two key components: Aspect-based KP extraction and Aspect-based KP Matching. First, leveraging the fine-grained aspect-based sentiment analysis (ABSA) (Pontiki et al., 2016; Wan et al., 2020; Zhang et al., 2021; Miao et al., 2020), ABKPA extracts KPs containing the opinion for a single aspect, free of redundancy. Next, leveraging ABSA

[1] A comment is a senence in reviews

Table 1: An example showing the summary output of ABKPA and sentence-based KPA (Bar-Haim et al., 2021). Given (a) The input comments, we exemplify and compare the output of (b) sentence-based KPA and (c) ABKPA. In (b) and (c), the columns "Matched comments" and "Quantity" illustrate matching KPs to comments and quantifying KPs in the summary.

(a) **The input comments**. Each box represents a review containing several comments

Review	**Comments (review sentences)**
1	**1.1:** The service is great and the staff is friendly and engaging. **1.2:** The food is excellent but the portion is quite small and quite expensive.
2	**2.1:** The food has great taste but very small portion and the service is slow.
3	**3.1:** The service was good and the food was delicious. **3.2:** Staff is friendly and attentive.
4	**4.1:** Food was excellent and delicious. **4.2:** Service and staff are excellent.
...	...

(b) Sentence-based KPs and their salience score (Bar-Haim et al., 2021, 2020a) output. Note that a commment can only be matched with one KP on of highest confidence.

Key points	**Matched Comments**	**Salience score**
KP1: Service and staff are excellent. **KP2:** Service was prompt and friendly. (*redundant*) ...	1.1 3.1 ...	1 1 ...
KP3: Small and overpriced portion. **KP4:** Small food portion and slow service. (*redundant*) ...	1.2 2.1 ...	1 1 ...

(c) **ABKPA** KPs and their salience score. ABKPA ensures retrieving single-aspect key points with better opinion quantification specific to every comment's aspect

Key points	**Matched Comments**	**Salience score**
KP1: Food was excellent and delicious. **KP2:** Service was prompt and friendly. **KP3:** Staff is friendly and attentive. ...	1.2; 2.1; 3.1 1.1; 3.1 1.1 ...	3 2 1 ...
KP4: Small and overpriced portion. **KP5:** Service was poor and slow ...	1.2; 2.1 2.1 ...	2 1 ...

predictions for automatic annotation of silver labels for matching aspect-sentiment pairs, we design a contrastive learning model for better representation of opinions in KPs and comments, which leads to more accurate salience scores for quantifying KPs.

Table 1 presents a comparison between ABKPA and sentence-based KPA (Bar-Haim et al., 2020a, 2021). As an example, consider the long comment "2.1: The food has great taste but very small portion and the service is slow.". In Table 1b, sentence-based KPA, applying the supervised matching model at the sentence level, can only match this comment to *one* KP "KP4: Small food portion and slow service", missing the "great taste" opinion on the "food" aspect of the comment. On the other hand, ABKPA, leveraging fine-grained ABSA to perform KPA at the aspect level, can identify and match every opinion expressed on the "food" and "service" aspects of the comment to single-aspect KPs, "KP1", "KP4" and "KP5" correctly, as shown in Table 1c. Nevertheless, with both comments and KPs containing opinions on multiple aspects, sentence-based KPA also becomes ineffective and results in inaccurate KP prevalence. For instance, in Table 1b, sentence-based KPA falsely map comment "1.1" and "3.1" with two overlapping KPs: "KP1" and "KP2", while both contain duplicate opinions on the same "service" aspect.

Our main contributions are: **(1)** We propose Aspect-based Key Point Analysis (ABKPA), a novel summarization framework for reviews. ABKPA addresses the KPA shortcomings in sentence-based KP extraction and matching, which extract KPs with overlapping opinions and falsely match KPs to long review comments containing multiple opinions. **(2)** Core to ABKPA is the use of a fine-grained ABSA model to extract aspect-focused KPs without redundancy. **(3)** Importantly, using fine-grained ABSA tagging to automatically generate and annotate silver labels for aspect-sentiment matching examples, we employed contrastive learning and devised an aspect-based KP Matching model for more accurate KP quantification on reviews.

2 Related Work

Based on the form of summaries, review summarization studies can be broadly grouped into three classes: aspect-based structured summarization, textual summarization, and key point analysis.

2.1 Aspect-based Structured Summarization

Early studies in the Data Mining community applied aspect-based sentiment analysis (ABSA) to extract, aggregate, and quantify opinions in reviews in the form of noun phrases (e.g., food, price, service) and positive and negative sentiment of the reviewed entity (Hu and Liu, 2004; Ding et al., 2008; Popescu and Etzioni, 2007; Blair-Goldensohn et al., 2008; Titov and McDonald, 2008). While these studies give basic quantification for reviews in terms of aspects and their sentiment, they lack textual explanation for the opinion details.

2.2 Textual Summarization

Document summarization is an important topic in the Natural Language Processing community, aiming to produce concise textual summaries capturing the salient information in source documents. While extractive review summarization approaches use surface features to rank and extract salient opinions for summarization (Mihalcea and Tarau, 2004; Angelidis and Lapata, 2018; Zhao and Chaturvedi, 2020), abstractive techniques use sequence-to-sequence models (Chu and Liu, 2019; Suhara et al., 2020; Bražinskas et al., 2020b,a; Zhang et al., 2020) to generate review-like summaries containing only the most prevalent opinions. Recently, prompted opinion summarization leveraging Large Language Models (LLMs) was applied to generate fluent and concise review summaries (Bhaskar et al., 2023). Still none of the existing studies focus on presenting and quantifying the diverse opinions in reviews.

2.3 Key Point Analysis

Originally developed to summarize arguments (Bar-Haim et al., 2020a,b), KPA was later applied to summarize and quantify the prevalence of opinions in reviews (Bar-Haim et al., 2021). Existing work on KPA for reviews has two major shortcomings. First, extraction of KPs relies on supervised models to identify short sentences with high argument quality as KPs, and such sentence-based extraction makes KPs often contain multiple and redundant opinions. Secondly, due to supervised training for the comment-KP matching model, despite containing multiple opinions, each comment is often mistakenly matched to a KP, leading to inaccurate quantification for KPs.

More recent research aims to generate high-level abstractive summaries for KPA. One class of studies (Cattan et al., 2023) is focused on structuring the KPs from extractive KPA as a hierarchy. Another class of studies is focused on abstractive summarization for KP generation (Kapadnis et al., 2021; Li et al., 2023b); an abstractive summarization model is employed to generate KPs either from each argument (Kapadnis et al., 2021), or by summarizing a cluster of arguments grouped by a common theme (Li et al., 2023b). None of the recent studies focus on the core issues of KP redundancy KPs and inaccurate quantification for KPs.

3 Aspect-based Key Point Analysis

As discussed earlier, there are two core issues in the current KPA studies, namely redundant KPs and inaccurate quantification for KPs. To address these two issues, we propose the ABKPA framework for aspect-based key point analysis of reviews. Figure 1 illustrates the training and inference stages of our ABKPA framework. ABKPA mainly leverages aspect-based sentiment analysis (ABSA; (Pontiki et al., 2016; Wan et al., 2020; Zhang et al., 2021) for Aspect-based KP Extraction of KPs with distinct aspects (Section 3.1) and aspect-based KP Matching (Section 3.2) for more effective comment-KP matching through contrastive learning for more effective fine-grained opinion representations. Notably, to bootstrap contrastive learning, we employ ABSA to automatically annotate aspect-sentiment pairs with silver labels for matching (Section 3.3).

3.1 Aspect-based KP Extraction

We address the issue of redundant opinions in KPs from short comments through aspect-based KP extraction, leveraging fine-grained BSA models. Existing studies on ABSA (Pontiki et al., 2016; Wan et al., 2020; Zhang et al., 2021) produce prediction labels for elements such as aspects and sentiment (positive or negative). We employ the four elements from the (a, c, o, s) quadruple prediction of ABSA (Zhang et al., 2021), namely (a)spect term, (c)ategory for the aspect, (o)pinion term and (s)entiment, to achieve KP extraction.

Figure 2 illustrates the ABSA predictions, where (a) is the aspect (e.g., *food*, *service*) of the entity under review, on which users express their opinion (o), while (c) generalizes (a) into categories (e.g., *FOOD_QUALITY*, *SERVICE*), and (s) indicates the sentiment for (o), that is *+ve*, or *-ve*.

To achieve aspect-based KP extraction, we start with collecting high-quality KP candidates using the argument quality ranking model from (Bar-

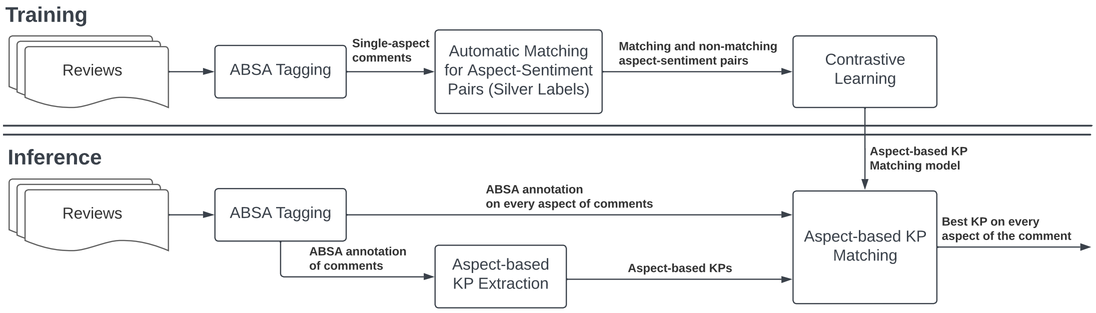
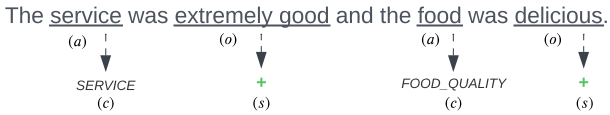
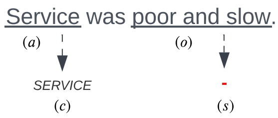


Figure 1: The training and inference phases of the ABKPA framework



(a) (a, c, o, s) elements of the comment: "The service was extremely good and the food was delicious.". The comment contains two opinions $(service, SERVICE, extremely\ good, +ve)$ and $(food, FOOD\_QUALITY, delicious, +ve)$, and therefore is not selected as a KP.



(b) (a, c, o, s) elements of the comment: "Service was poor and slow.". The comment contains only one opinion $(service, SERVICE, poor\ and\ slow, -ve)$, and therefore is selected as a KP.

Figure 2: Elements of the quadruple prediction (a, c, o, s) of ABSA for two example comments taken from Table 1. The examples also illustrate that aspect-based KP selection only selects KPs for single aspects.

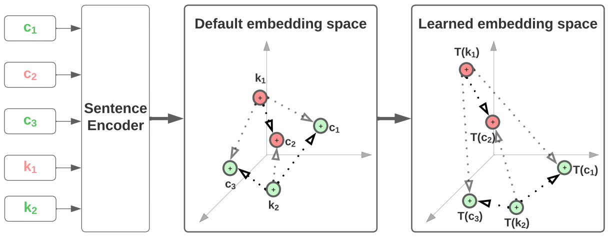


Figure 3: An example of the opinion embedding space transformation through contrastive learning. In this example, each node represents the opinion on a particular aspect of a comment (c) or key point (k), and is colored by their sentiments. The positive pairs (e.g., k_1 and c_2), whose (a, o, s) triplet of the opinions share a great similarity, are pulled closer to each other while negative pairs are pushed apart.

Haim et al., 2021), before performing ABSA prediction to retrieve the opinion phrases of all KP candidates. Then, we select only KPs having a single aspect and opinion, and sort KPs by descending order of their quality. Finally, we traverse the candidates from the list, targeting overlapping KPs with identical (a, o, s) triplet, and remove those with higher length yet lower quality from the list.

3.2 Aspect-based KP Matching Using Contrastive Learning

We devise an aspect-based KP matching model for ABKPA, which directly scores the similarity of a single opinion of a comment for a KP. As illustrated in Figure 3, aspect-based KP matching employs contrastive learning to transform the original semantic embedding of a comment or KP into a new space where the position of positive matching pairs - with signals indicated by the (a, o, s) triplet of an opinion in comments and KPs - are closer than negative pairs, and vice versa.

We utilize the siamese neural network architecture, which was proven efficient for encoding sentences (Reimers and Gurevych, 2019), for training the aspect-based KP matching model. Formally, considering a single opinion from a comment (c) and key point (k), we create the training input as $\{T(\mathrm{c}), T(\mathrm{k}), label\}$, where $T(\mathrm{c})$ or $T(\mathrm{k})$ uses a special token <SEP> to concatenate tokens of the (a, o, s) triplet of an opinion from c or k, and $label$ is the label for matching aspect-sentiment pairs, where 0 indicate a non-matching (negative) pair and 1 indicates a matching pair (positive). An example of $T(\mathrm{c})$ or $T(\mathrm{k})$, taken as the opinion of a comment from Table 1, is "friendly and attentive staff <SEP> positive". We then use a pre-trained language model to encode tokens in $T(\mathrm{c})$ and $T(\mathrm{k})$ of the pair. Then, we pass their embeddings through a siamese neural network, which is a mean-pooling layer to aggregate the token embeddings of each input into sentence embeddings. We compute the contrastive loss of sentence embeddings of each training input as:

$$\mathcal{L} = -y \cdot \log(\hat{y}) + (1-y) \cdot \log(1-\hat{y}) \quad (1)$$

where $\hat{y}$ is the cosine similarity of the embeddings, and y reflects whether a pair matches (1) or not (0). Using contrastive loss (Equation 1), the network is trained to encode the input sequences to make positive and negative examples more distinguishable in the new embedding space. During inference, sequences of single opinions from the comment-KP pairs are input into the network, and the cosine similarity is used to compute their matching score.

Because our new aspect-based KP matching model utilizes the aspect-sentiment predictions, it also allows matching opinions for multiple aspects to multiple key points for the same aspect, which is more accurate than matching at the sentence level in existing KPA studies (Bar-Haim et al., 2020b, 2021). During inference, given a comment and a set of aspect-based KPs, we first calculate the matching scores of opinions inside comments with all KPs as the cosine similarity for their aspect-sentiment-based opinion representation space. We then match every opinion to its best-matching KP.

As discussed earlier, to achieve effective contrastive learning for the aspect-based KP matching model, comment-KP pairs annotated with positive (matching) and negative (non-matching) labels are needed for training the model. We next present our approach to leveraging ABSA predictions to automatically construct such training examples with silver labels for matching.

3.3 Automatic Annotation of Silver Labels for Matching Aspect-Sentiment Pairs

The positive (matching) and negative (non-matching) aspect-sentiment pairs are crucial to train the opinion embedding space of KPs and comments for our aspect-based KP Matching model. We employ the ABSA predictions to automatically annotate aspect-sentiment pairs with positive (matching) or negative (non-matching) labels. These labels are silver labels (Amplayo et al., 2021) as they are derived from ABSA automatic predictions and may not be fully correct. Nevertheless, our experiments show that a reasonably large number of examples with silver labels (in our case 600-2000 examples) are sufficient to train an effective model.

We next explain the details for annotating matching aspect-sentiment pairs based on ABSA predictions for sentences. Given ABSA prediction triplet (a, c, s) – (a)spect term and its (c)ategory, and (s)entiment – for a pair of sentences, we give the positive label to a pair of sentences if they have the same sentiment, aspect category, and the cosine similarity for their aspect terms are above a threshold (determined empirically). Specifically,

$$c(\mathrm{c}) = c(\mathrm{k}),\ cos(\mathbf{e}^{a(\mathrm{c})}, \mathbf{e}^{a(\mathrm{k})}) \geq \theta,\ s(\mathrm{c}) = s(\mathrm{k})$$

where c and k are the pair of sentences, $c(\mathrm{c})$ and $c(\mathrm{k})$ are the aspect categories from c and k, $\mathbf{e}^{a(\mathrm{c})}$ and $\mathbf{e}^{a(\mathrm{k})}$ are the word embeddings of aspect terms from c and k, $s(\mathrm{c})$ and $s(\mathrm{k})$ are the sentiments from c and k, respectively, and $\theta \in (0, 1]$ is a threshold for deciding the homogeneity of the pair's aspect terms. We compute the cosine similarity for the pair's aspect terms as:

$$\cos(\mathbf{e}^{a(\mathrm{c})}, \mathbf{e}^{a(\mathrm{k})}) = \frac{\mathbf{e}^{a(\mathrm{c})^T} \mathbf{e}^{a(\mathrm{k})}}{||\mathbf{e}^{a(\mathrm{c})}||_2 \, ||\mathbf{e}^{a(\mathrm{k})}||_2} \quad (2)$$

Note that the above approach to generating matching aspect-sentiment pairs implies that only sentences containing single aspects are used to construct training examples. We label the remaining pairs disqualified by the above matching criteria as negative pairs whose opinions have dissimilar aspects and/or sentiments.

4 Experiments

4.1 Experiment Setup

We compared the KP matching performance of ABKPA against the following state-of-the-art models:

RKPA: The latest sentence-based KP Matching model for reviews (Bar-Haim et al., 2021). The supervised KP matching model was trained using ArgKP, a KP Matching dataset for arguments (Bar-Haim et al., 2020a).

RKPA+: An enhanced version of RKPA (Bar-Haim et al., 2021), where RKPA is fine-tuned using our aspect-sentiment matching examples with silver labels for training the KP matching model. We use this baseline to evaluate the effectiveness of silver-annotated training examples.

SMatch: A model based on the first-ranked KP matching model for arguments from the KPA-2021 shared task (Friedman et al., 2021). We further fine-tuned it using our aspect-sentiment matching examples with silver labels for training the KP matching model. Note that SMatch employs contrastive learning to model the cosine similarity of comments and KPs based on the embedding of their whole sentences. We use SMatch to evaluate the effectiveness of contrastive learning in ABKPA, utilize aspect-sentiment annotations to specifically measure the cosine similarity of opinions in the comment-KP pairs.

Note that conventionally, RKPA, RKPA+, and SMatch can only match a comment to one best-matching KP, which makes them always fail to match a comment of multiple opinions with multiple KPs. For fair comparison, we adjust these models to match each comment with top n highest-scored KPs, where n is the number of opinion aspects in the comment.

ABKPA, together with the baseline models, were all fine-tuned on a RoBERTa-large model (Liu et al., 2019), using the Huggingface transformers framework. For hyperparameters for all baseline models, we used the optimal setting reported in previous studies for their best performance. We first pretrained all models with the Masked LM (MLM) task (Liu et al., 2019) to adapt it to reviews. The pretraining was performed for 2 epochs, a learning rate of 1e-5, following the procedure described by Bar-Haim et al. (2021). For ABKPA and SMatch, based on the setting of Alshomary et al. (2021), we fine-tuned the siamese network of the model for 10 epochs, with a batch size of 16, and a maximum input length of 128 , leaving all other parameters to their defaults. For RKPA and RKPA+, we fine-tuned the KP Matching model for 9 epochs, with a learning rate of 5e-6, as suggested by (Bar-Haim et al., 2021) , keeping all other settings at their default values. We trained all models using an NVIDIA GeForce RTX 3080Ti GPU. We implemented the model Snippext (Miao et al., 2020) to obtain ABSA predictions on review comments. For annotation of silver labels for matching sentence pairs, we employ Spacy (Honnibal et al., 2020) to compute the cosine similarity for their aspect terms.

4.2 Data

Our experiments used the popular Yelp Open Dataset [2], consistent with the latest KPA work (Bar-Haim et al., 2021), but we extended to reviews for five business categories: *Arts & Entertainment* (25k reviews), *Automotive* (41k reviews), *Beauty & Spas* (72k reviews), *Hotels* (8.6K reviews), and *Restaurants* (680k reviews).

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

Table 2: Annotations for test data in five dataset (i.e, business categories): Arts (& Entertainment), Auto(motive), Beauty (& Spas), Hotels, Restaurants.

Dataset	# pairs	# +ve pairs	# KPs
Arts	1536	69	32
Auto	877	93	18
Beauty	1093	77	22
Hotels	1680	72	35
Restaurants	1613	108	33

Each dataset, corresponding to a specific business category, was divided into ’training’ and ’test’ subsets. Reviews from the first and second top 30 most-commented business entities were sampled for training and test, respectively. In this way we ensure that there are not overlapping business entities between the training and test data. For both training and test subsets, we extract aspect-based KP candidates, constrained to 3-6 tokens, first following Bar-Haim et al. (2021) to compute the quality score of comments using the argument quality model (Toledo et al., 2019), with the minimum quality score 0.42.

In the test subsets, for annotating the matching ground truth in test data (for evaluation), we used the Amazon Mechanical Turk [3] (MTurk) as the crowdsourcing platform for manual annotation, based on the guideline of Bar-Haim et al. (2020a) and Bar-Haim et al. (2021). We collected labels from 8 annotators for each matching pair. To ensure annotation quality, we only selected answers from annotators with high agreement with others, where minimum κ score is 0.05. Details for the annotation scheme and quality control to ensure high-quality annotation are in Appendix A.

Table 2 summarises the statistics of the test data and their annotations for each of the five business categories. Overall, the test dataset has 6799 labelled (comment, KP) pairs, of which 419 pairs are positive.

4.3 Results

We fine-tuned all models on the training subset and evaluated them on the test subsets for different business categories, except for RKPA, which was fine-tuned on ArgKP following the implementation of Bar-Haim et al. (2021); each category can be seen as a dataset. Our evaluation used the metric Average Precision (AP), the same as in the *KPA-*

[3]https://www.mturk.com/

Table 3: AP score of KP Matching models. The best result of each experiment is in bold.

Dataset	All comments				Multiple-opinion comments			
	ABKPA	**SMatch**	**comm-Match**	**RKPA**	**ABKPA**	**SMatch**	**comm-Match**	**RKPA**
Arts	**0.99**	0.98	0.94	0.79	**0.99**	0.88	0.83	0.90
Auto	**0.77**	0.75	0.43	0.54	**0.80**	0.70	0.42	0.71
Beauty	**0.98**	0.97	0.84	0.62	**0.94**	0.88	0.81	0.62
Hotels	**0.99**	0.98	0.98	0.81	**0.93**	0.89	0.93	0.81
Restaurants	**0.87**	0.85	0.73	0.50	**0.83**	0.75	0.73	0.56
Average	**0.92**	0.91	0.78	0.65	**0.90**	0.82	0.74	0.72

Table 4: Model generalizability evaluation results. AP score in *out-of-category* experiment of KP Matching models, where data for one category is used for testing and models are trained on data for the rest categories. Note that no results for RKPA as it is trained on non-Yelp review data. The best result of each experiment is in bold. Result difference from the within-category experiment (Table 3) is shown in brackets, while (—-) indicates nil difference.

Dataset	All comments			Multiple-opinion comments		
	ABKPA	**SMatch**	**RKPA+**	**ABKPA**	**SMatch**	**RKPA+**
Arts	**0.98** (-.01)	0.95 (-.03)	0.90 (-.04)	**0.99** (—-)	0.80 (-.08)	0.83 (—-)
Auto	**0.76** (-.01)	0.51 (-.24)	0.40 (-.03)	**0.64** (-.12)	**0.64** (-.08)	0.41 (-.01)
Beauty	**0.94** (-.04)	0.97 (—-)	0.60 (-.24)	0.77 (-.17)	**0.84** (-.04)	0.54 (-.27)
Hotels	**0.98** (-.01)	0.96 (-.02)	0.92 (-.06)	**0.92** (-.01)	0.81 (-.07)	0.89 (-.04)
Restaurants	**0.87** (—-)	0.84 (-.01)	0.66 (-.07)	**0.75** (-.08)	0.61 (-.14)	0.69 (-.04)
Average	**0.91** (-.01)	0.85 (-.06)	0.70 (-.09)	**0.81** (-.08)	0.74 (-.08)	0.67 (-.04)

2021 shared task (Friedman et al., 2021) [4]. First, for all models, we extract the top 50% predicted matching pairs for each dataset by the order of their confidence (matching) score. Then, given the ground truth data, Average Precision (Turpin and Scholer, 2006) (AP), is calculated per dataset to evaluate the model matching performance. During evaluation, models are tested on two data configurations: "all comments" and "multiple-opinion comments", which explicitly aim to test the model's ability to handle comments with multiple opinions.

Table 3 presents the AP score for all models under "all comments" or "multiple-opinion comments" configurations. Overall, ABKPA shows the best performance, significantly outpacing other models (paired t-test, $p << 0.05$), with an average AP score of 0.92 and 0.90. Conversely, RKPA shows the lowest performance in three out of five datasets, mainly because it was fine-tuned with argument data and applied to reviews. RKPA+, sharing RKPA architecture but was fine-tuned using our silver-annotated reviews, displays a higher performance overall. Finally, SMatch and ABKPA, by applying contrastive learning for KP Matching on the natural content of comments or on the opinion information of comments, respectively, achieve consistent improvements on all datasets. While both alternatives perform well and apply contrastive learning, ABKPA achieves higher and more consistent performance. This again demonstrates the benefit of integrating ABSA resources into ABKPA's KP Matching task.

In the "multiple-opinion comment" scenario, most models saw a certain performance decrease, mainly due to the long comments of multiple opinions challenging KP Matching. Surprisingly, RKPA shows a slight performance boost, likely benefiting from its extensive training data with longer sentences from the argument domain compared to our silver-annotated data. However, ABKPA still maintains its leading position with minimal performance variation.

4.4 Out-of-category experiment

In this set of experiments, we assess the generalizability of ABKPA and baseline models via out-of-category performance evaluation. Specifically, we test each model's performance on a dataset with a business category c (e.g., hotels), considering it was trained on all other datasets excluding c.

[4] https://2021.argmining.org/shared_task_ibm

Table 5: AP score of ABKPA and ABKPA$_{\neg C}$ on two test data settings.

Dataset	All comments		Multi-opinion comments	
	ASK-PA	**ASK-PA$_{\neg C}$**	**ASK-PA**	**ASK-PA$_{\neg C}$**
Arts	0.99	0.92	0.99	0.89
Auto	0.77	0.58	0.80	0.43
Beauty	0.98	0.85	0.94	0.82
Hotels	0.99	0.95	0.93	0.88
Restaurants	0.87	0.78	0.83	0.72

Table 4 presents the AP Score for all models in the out-of-category experiment. Comparing Table 3 and Table 4, the relative ranking of models remains similar, with ABKPA showing the best and most stable performance. In the "all comments" setting, ABKPA shows a very slight decrease in its AP Score (0.1 on average, drop varying from 0.01 to 0.04), while still outperforming other models significantly (paired t-test, $p < 0.05$), with an average AP score of 0.91. This shows that ABKPA can be generalized to new, unseen business categories. In contrast, SMatch and RKPA+ see notable performance drops – 0 to 0.24 for SMatch and 0.03 to 0.24 for RKPA+ – when transitioning from in-category to out-of-category, indicating their domain dependence, a finding aligned with existing studies. For multi-opinion comments, ABKPA remains the top performer with an AP score of 0.81 (compared to 0.74 for SMatch and 0.67 for RKPA+), while RKPA+ sees the most significant drop – from 0.04 to 0.27, emphasizing the instability of domain-dependent supervised training models.

4.5 Ablation study

Our ablation study examines the utility of contrastive learning in KP Matching. The ABKPA$_{\neg c}$ model, omitting constrastive learning, uses the positive and negative examples from our silver-annotated data to directly train a matching model. Table 5 highlights the performance disparity between ABKPA$_{\neg c}$ and ABKPA. Without contrastive learning, ABKPA$_{\neg c}$ exhibits a significant performance decline, highlighting the efficacy of contrastive learning in ABKPA. In the “all comments” setting, the average absolute AP score decreases by 0.10, ranging from 0.04 to 0.19. For ”multi-opinion comments”, the performance drop of ABKPA$_{\neg c}$ is even more pronounced, with the AP score declining from 0.90 to 0.75, varying from 0.05 to 0.37. These results demonstrate the importance of contrastive learning for the superb performance of ABKPA.

4.6 Case studies

We conduct a case study to evaluate KP redundancy on the “Restaurants” dataset, as shown in Table 7 (Appendix D). Overall, all baselines encounter redundancy (i.e., KPs with overlapping aspects and opinions) in the output. For example, for the baseline model RKPA+, the KP “Customer service is excellent.” contain redundant positive opinion on service with the KP “The service here was exceptional”. In contrast, ABKPA offers KP matching with distinct, diverse aspects in comments.

We conduct another case study to evaluate the correctness of KP prevalence (i.e., salience score) of different models on popular KPs (i.e., KPs with a high number of matching comments in the ground truth annotations). Table 8 (Appendix E) presents the prevalence quantity, or salience score, for KPs by each model for the top three most prevalent KPs from each dataset. Recall that ABKPA has the best matching performance among all models, as shown in Section 4.3. This table further shows that effective KP matching of ABKPA leads to its good performance for quantifying KPs. As can be seen from the table, overall all models show salience scores not very comparable to human annotations, and ABKPA shows the most stable performance compared to other models. ABKPA shows the lowest salience score for Automotive, due to its lowest matching performance for this category (as shown in Tables 3 and 4).

5 Conclusions

In this paper, we proposed a framework Aspect-Based Key Point Analysis, namely ABKPA, to address the issues of redundant opinions and inaccurate quantification for KPs in existing KPA studies. First, we leverage ABSA to extract KPs of distinct aspects, which significantly reduce KPs containing redundant opinions. Secondly, leveraging ABSA predictions, we automatically annotate matching aspect-sentiment for sentence pairs and achieve contrastive learning for effective fine-grained aspect-based opinion embeddings and aspect-based KP matching, leading to accurate quantification for KPs.

Limitations

The KP Matching model of ABKPA and other baselines was implemented using a RoBERTa large language model. Due to the high number of parameters (355M), the model requires high GPU resources for pre-training and fine-tuning. With limited GPU resource, we restrict the maximum input length of the baseline models to be 512 tokens. Our development, utilization of language model, and reported performance assume the framework to suitably be implemented for English.

Ethics Statement

We have applied ethical research standards in our organization for data collection and processing throughout our work.

The Yelp dataset used in our experiments was officially released by Yelp, which was published by following their ethical standard, after removing all personal information. The summaries do not contain contents that are harmful to readers.

We ensured fair compensation for crowd annotators on Amazon Mechanical Turk. We setup and conducted fair payment to workers on their annotation tasks/assignments according to our organization's standards, with an estimation of the difficulty and expected time required per task based on our own experience. Especially, we also made bonus rewards to annotators who exerted high-quality annotations in their assignments.

References

Milad Alshomary, Timon Gurcke, Shahbaz Syed, Philipp Heinisch, Maximilian Spliethöver, Philipp Cimiano, Martin Potthast, and Henning Wachsmuth. 2021. Key point analysis via contrastive learning and extractive argument summarization. In *Proceedings of the 8th Workshop on Argument Mining*, pages 184–189, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Reinald Kim Amplayo, Stefanos Angelidis, and Mirella Lapata. 2021. Aspect-controllable opinion summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6578–6593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3675–3686, Brussels, Belgium. Association for Computational Linguistics.

Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. 2020a. 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.

Roy Bar-Haim, Yoav Kantor, Lilach Eden, Roni Friedman, Dan Lahav, and Noam Slonim. 2020b. Quantitative argument summarization and beyond: Cross-domain key point analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 39–49, Online. Association for Computational Linguistics.

Adithya Bhaskar, Alex Fabbri, and Greg Durrett. 2023. Prompted opinion summarization with gpt-3.5. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9282–9300.

Sasha Blair-Goldensohn, Kerry Hannan, Ryan McDonald, Tyler Neylon, George Reis, and Jeff Reynar. 2008. Building a sentiment summarizer for local service reviews.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020a. Few-shot learning for opinion summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4119–4135, Online. Association for Computational Linguistics.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020b. Unsupervised opinion summarization as copycat-review generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5151–5169, Online. Association for Computational Linguistics.

Arie Cattan, Lilach Eden, Yoav Kantor, and Roy Bar-Haim. 2023. From key points to key point hierarchy: Structured and expressive opinion summarization. *arXiv preprint arXiv:2306.03853*.

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.

Abhisek Dash, Anurag Shandilya, Arindam Biswas, Kripabandhu Ghosh, Saptarshi Ghosh, and Abhijnan

Chakraborty. 2019. Summarizing user-generated textual content: Motivation and methods for fairness in algorithmic summaries. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–28.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240.

Roni Friedman, Lena Dankin, Yufang Hou, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2021. Overview of the 2021 key point analysis shared task. In *Proceedings of the 8th Workshop on Argument Mining*, pages 154–164, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Manav Kapadnis, Sohan Patnaik, Siba Panigrahi, Varun Madhavan, and Abhilash Nandy. 2021. Team enigma at ArgMining-EMNLP 2021: Leveraging pre-trained language models for key point matching. In *Proceedings of the 8th Workshop on Argument Mining*, pages 200–205, Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Hao Li, Viktor Schlegel, Riza Batista-Navarro, and Goran Nenadic. 2023a. Do you hear the people sing? key point analysis via iterative clustering and abstractive summarisation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14064–14080, Toronto, Canada. Association for Computational Linguistics.

Hao Li, Viktor Schlegel, Riza Batista-Navarro, and Goran Nenadic. 2023b. Do you hear the people sing? key point analysis via iterative clustering and abstractive summarisation. *arXiv preprint arXiv:2305.16000*.

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*.

Mary McGlohon, Natalie Glance, and Zach Reiter. 2010. Star quality: Aggregating reviews to rank products and merchants. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 4, pages 114–121.

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.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Ana-Maria Popescu and Orena Etzioni. 2007. Extracting product features and opinions from reviews. *Natural language processing and text mining*, pages 9–28.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Anurag Shandilya, Kripabandhu Ghosh, and Saptarshi Ghosh. 2018. Fairness of extractive text summarization. In *Companion Proceedings of the The Web Conference 2018*, pages 97–98.

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, Online. Association for Computational Linguistics.

Wenyi Tay, Xiuzhen Zhang, and Sarvnaz Karimi. 2020. Beyond mean rating: Probabilistic aggregation of star ratings based on helpfulness. *Journal of the Association for Information Science and Technology*, 71(7):784–799.

Ivan Titov and Ryan McDonald. 2008. A joint model of text and aspect ratings for sentiment summarization. In *proceedings of ACL-08: HLT*, pages 308–316.

Assaf Toledo, Shai Gretz, Edo Cohen-Karlik, Roni Friedman, Elad Venezian, Dan Lahav, Michal Jacovi, Ranit Aharonov, and Noam Slonim. 2019. Automatic argument quality assessment - new datasets and methods. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5625–5635, Hong Kong, China. Association for Computational Linguistics.

Andrew Turpin and Falk Scholer. 2006. User performance versus precision measures for simple search tasks. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 11–18.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9122–9129.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. 2021. Aspect sentiment quad prediction as paraphrase generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9209–9219, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chao Zhao and Snigdha Chaturvedi. 2020. Weakly-supervised opinion summarization by leveraging external information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9644–9651.

A Annotation and Labelling Details of Test Data

To prepare gold-labelled KPs in the test set for evaluation, we relied on human to annotate/select KPs. For each test subset, we guide annotators to select non-redundant KPs, prioritizing those with high-quality scores and fulfilling 4 properties of KPs for reviews (Bar-Haim et al., 2021), including *validity*, *sentiment*, *informativeness*, and *single-aspect* Similarly, to ensure consistent quality in the test subsets, we limit comments to a length of 6-11 tokens. For each token length in this range, we select the top 8 highest-quality comments, creating a total of 48 comments per category. We constructed the test data based on the above filtered comments and aspect-based KPs.

For labelling the matching pairs on the test data for evaluation, we mainly annotate data using the Amazon Mechanical Turk [5] (MTurk) crowdsource platform, based on the guidelines of Bar-Haim et al. (2020a) and Bar-Haim et al. (2021). To ensure annotation quality, we only select workers with $\geq$ 80% lifetime approval rate and have at least 10 annotations approved). For each comment, annotators were prompted to select none or multiple relevant key points, where they are not exposed to any ABSA information to ensure fair evaluation of all models and not to favour ABKPA. Note also that each comment was labeled by 8 annotators, and they can freely decide the number of matching key points to a comment. Further, following Bar-Haim et al. (Bar-Haim et al., 2021), we ignore the judgement of annotators whose annotator-κ score < 0.05. This score averages all pair-wise Cohen's Kappa (Landis and Koch, 1977) for a given annotator, for any annotator sharing at least 50 judgments with at least 5 other annotators. Details of the annotation task description and guidelines for the crowd-workers are provided in Appendix B.

We consolidate the labels for every matching pair following Bar-Haim et al. (Bar-Haim et al., 2020a), where the *agreement score* for a comment-KP pair – the fraction of annotations as matching – is used to select positive and negative pairs. We decided to label comment-KP pair as **(i)** positive if the agreement score $> 30\%$, **(ii)** negative if agreement score $< 15\%$; and **(iii)** otherwise undecided. Note that there are no undecided pairs because the annotation covers the labels for all possible pairs. Note also that the agreement score threshold of 30% for labelling positive pairs is different from the 60% threshold used for argument data by Bar-Haim et al. (Bar-Haim et al., 2020a)) and is set empirically. Details of the experiment are provided in Appendix C.

[5] https://www.mturk.com/

B Key Point Matching Annotation Guideline of Test Data

We report details of the annotation task description and instruction to the Amazon Mechanical Turk crowd-workers as follows:

Task title: Match the review sentence to its relevant key point(s)

Task description: Workers are required to mark valid key point(s) (short, high-quality, and concise sentences) that represent the content of a sample sentence

Instruction:

In this task you are presented with a business domain, a sentence taken from a review of a business in that domain and a key point.

Choose multiple key points that represent the content (of mentioned aspects) in the given sentence.

Note that a sentence might cover opinions on multiple aspects of the reviewed entity. Please select all relevant KPs that represent all aspects mentioned in the sentence.

C Analysis of Agreement Score for Positive Label on Test Data Annotation

We use an agreement score threshold of 30% for labelling positive pairs for reviews, different than the 60% used for argument data by Bar-Haim et al. (2020a)). For reviews, because sentences are shorter and are more likely to contain overlapping opinions than online argument debates, annotators tend to select more KPs to match a comment. For example, the annotators might match the comment *"waitress was very polite"* to either or both *"staff is courteous"*, and *"servers are great"* key points, and have less consistent annotations. Table 6 shows the percentage of comments by key point matches using different thresholds t for the agreement score within 0.1-0.6. In this measurement, a comment is matched to a key point if at least t annotators agree. Similarly, a comment has no key point if at least t annotators match it to 'None'. Otherwise, the comment is 'ambiguous'. From Table 6, we observe a tradeoff between the number of positive comment-KP pairs and the agreement score. As soon as the agreement score threshold is above 0.3, there are more comments with insufficient confidence in their annotations while matching with key points, resulting in a high proportion of ambiguous cases. We, therefore, use 0.3 as the threshold for the agreement score. Interestingly, from Table 6, key points selected by humans can cover about 90% of comments, with 50.83% of the comments mapped to more than one key point, showing the quality of our annotation for comments with multiple aspects.

D KP Summary Output

This section presents details of Table 7, which shows the top 5 negative KPs for all models, ranked by their prevalence, for the Hotels domain,

E KP Matching Prevalence Output

This section presents details of Table 8, which shows the performance of different models in our case study on the top three important KPs in every dataset.

Table 6: Percentage of comments by key point matches by different agreement score for matching pairs

Agreement score	**No key point**	**Ambiguous**	**Single KP**	**Multiple KP**
0.1	0.42%	0%	2.08%	97.50%
0.2	2.08%	0%	20.83%	77.08%
0.3	**5.83%**	**3.33%**	**40.00%**	**50.83%**
0.4	6.25%	13.75%	53.75%	26.25%
0.5	6.25%	13.75%	53.75%	26.25%
0.5	2.08%	35.42%	53.75%	8.75%

Table 7: Top 6 positive-sentiment key points ranked by their predicted prevalence on "Restaurants" datasets. While ABKPA generates distinct KPs on single aspects, baseline models generate KPs with overlapping aspects and opinions. KPs that overlap with higher-ranked ones (i.e., KPs with higher prevalence) are noted with a (*redundant*) postfix

ABKPA	**SMatch**	**RKPA+**	**RKPA**	**ABKPA**$_{\neg C}$
Staff was courteous and accommodating.	Staff was courteous and accommodating.	Staff was courteous and accommodating.	Employees are friendly and attentive.	Staff was courteous and accommodating.
Generous sized portions.	Prices are fair and reasonable.	The service here was exceptional.	The service here was exceptional.	Fresh food , using local produce.
Service was prompt and friendly.	Fresh food , using local produce.	Fresh food , using local produce.	Ambiance is casual and comfortable.	Customer service is excellent.
Fantastic drink selection.	The service here was exceptional.	The food is consistently excellent!	Fresh food , using local produce.	The service here was exceptional. (*redundant*)
Prices are fair and reasonable.	Generous sized portions.	Customer service is excellent. (*redundant*)	Really delicious food , well balanced!	Lots of outdoor seating.
Delicious and expertly prepared food.	Service was prompt and friendly. (*redundant*)	Prices are fair and reasonable.	Staff was courteous and accommodating. (*redundant*)	Amazing authentic flavor!

Table 8: Prevalence on important key points (top three most common KPs among the framework) comparing with the ground truth.

#	Key Point	ABKPA	SMatch	comm-Match	RKPA	AS-KPA$_{\neg c}$	Human
Arts (& Entertainment)							
1	Friendly and helpful staff.	10	10	12	10	10	14
2	Seats are adequately comfortable.	4	6	4	5	4	4
3	Horrible customer service.	2	3	2	3	3	3
Auto(motive)							
1	They have excellent customer service.	6	7	1	4	10	29
2	The employees here are wonderful!	3	2	1	12	2	13
3	Very professional staff	4	5	3	2	0	13
Beauty (& Spas)							
1	Staff is friendly and accomodating.	14	14	33	6	13	18
2	Customer service- Excellent!	5	5	4	2	7	13
3	Amazing & professional service.	3	1	4	24	3	14
Hotels							
1	Friendly and helpful staff.	19	15	16	19	16	21
2	Clean and comfortable rooms.	9	10	8	11	12	13
3	The ambiance is wonderfully peaceful	1	2	3	0	2	1
Restaurants							
1	Staff was courteous and accomodating.	10	12	10	3	11	19
2	Fresh food, using local produce.	5	5	7	3	8	5
3	The service here was exceptional	2	5	6	6	5	5