OPINIONDIGEST: A Simple Framework for Opinion Summarization

Yoshihiko Suhara∗1, Xiaolan Wang∗1, Stefanos Angelidis2, Wang-Chiew Tan1
1Megagon Labs 2University of Edinburgh
{yoshi,xiaolan,wangchiew}@megagon.ai s.angelidis@ed.ac.uk

Abstract

We present OPINIONDIGEST, an abstractive opinion summarization framework, which does not rely on gold-standard summaries for training. The framework uses an Aspect-based Sentiment Analysis model to extract opinion phrases from reviews, and trains a Transformer model to reconstruct the original reviews from these extractions. At summarization time, we merge extractions from multiple reviews and select the most popular ones. The selected opinions are used as input to the trained Transformer model, which verbalizes them into an opinion summary. OPINIONDIGEST can also generate customized summaries, tailored to specific user needs, by filtering the selected opinions according to their aspect and/or sentiment. Automatic evaluation on YELP data shows that our framework outperforms competitive baselines. Human studies on two corpora verify that OPINIONDIGEST produces informative summaries and shows promising customization capabilities.

1 Introduction

The summarization of opinions in customer reviews has received significant attention in the Data Mining and Natural Language Processing communities. Early efforts (Hu and Liu, 2004a) focused on producing structured summaries which numerically aggregate the customers’ satisfaction about an item across multiple aspects, and often included representative review sentences as evidence. Considerable research has recently shifted towards textual opinion summaries, fueled by the increasing success of neural summarization methods (Cheng and Lapata, 2016; Paulus et al., 2018; See et al., 2017; Liu and Lapata, 2019; Isonuma et al., 2019).

Opinion summaries can be extractive, i.e., created by selecting a subset of salient sentences from the input reviews, or abstractive, where summaries are generated from scratch. Extractive approaches produce well-formed text, but selecting the sentences which approximate the most popular opinions in the input is challenging. Angelidis and Lapata (2018) used sentiment and aspect predictions as a proxy for identifying opinion-rich segments. Abstractive methods (Chu and Liu, 2019; Brainskas et al., 2019), like the one presented in this paper, attempt to model the prevalent opinions in the input and generate text that articulates them.

Opinion summarization can rarely rely on gold-standard summaries for training (see Amplayo and Lapata (2019) for a supervised approach). Recent work has utilized end-to-end unsupervised architectures, based on auto-encoders (Chu and Liu, 2019; Brainskas et al., 2019), where an aggregated representation of the input reviews is fed to a decoder, trained via reconstruction loss to produce review-like summaries. Similarly to their work, we assume that review-like generation is appropriate for opinion summarization. However, we explicitly deal with opinion popularity, which we believe is crucial for multi-review opinion summarization. Additionally, our work is novel in its ability to explicitly control the sentiment and aspects of selected opinions. The aggregation of input reviews is no longer treated as a black box, thus allowing for controllable summarization.

Specifically, we take a step towards more interpretable and controllable opinion aggregation, as we replace the end-to-end architectures of previous work with a pipeline framework. Our method has three components: a) a pre-trained opinion extractor, which identifies opinion phrases in reviews; b) a simple and controllable opinion selector, which merges, ranks, and optionally filters the extracted opinions; and c) a generator model, which is trained...
to reconstruct reviews from their extracted opinion phrases and can then generate opinion summaries based on the selected opinions.

We describe our framework in Section 2 and present two types of experiments in Section 3: A quantitative comparison against established summarization techniques on the YELP summarization corpus (Chu and Liu, 2019); and two user studies, validating the automatic results and our method’s ability for controllable summarization.

2 OPINIONDIGEST Framework

Let $D$ denote a dataset of customer reviews on individual entities $\{e_1, e_2, \ldots, e_{|D|}\}$ from a single domain, e.g., restaurants or hotels. For every entity $e$, we define a review set $R_e = \{r_i\}_{i=1}^{|R_e|}$, where each review is a sequence of words $r = (w_1, \ldots, w_n)$.

Within a review, we define a single opinion phrase, $o = (w_{a1}, \ldots, w_{a|O|})$, as a subsequence of tokens that expresses the attitude of the reviewer towards a specific aspect of the entity. Formally, we define the opinion set of $r$ as $O_e = \{(o_i, pol_i, a_i)\}_{i=1}^{O_e}$, where $pol_i$ is the sentiment polarity of the $i$-th phrase (positive, neutral, or negative) and $a_i$ is the aspect category it discusses (e.g., a hotel’s service, or cleanliness).

For every entity $e$, our task is to abstractly generate a summary $s_e$ of the most salient opinions expressed in reviews $R_e$. Contrary to previous abstractive methods (Chu and Liu, 2019; Brainskas et al., 2019), which never explicitly deal with opinion phrases, we put the opinion sets of reviews at the core of our framework, as described in the following sections and illustrated in Figure 1.

2.1 Opinion Extraction

Extracting opinion phrases from reviews has been studied for years under the Aspect-based Sentiment Analysis (ABSA) task (Hu and Liu, 2004b; Luo et al., 2019; Dai and Song, 2019; Li et al., 2019).

We follow existing approaches to obtain an opinion set $O_e$ for every review in our corpus.

Specifically, we used a pre-trained tagging model (Miao et al., 2020) to extract opinion phrases, their polarity, and aspect categories. Step 1 (top-left) of Figure 1 shows a set of opinions extracted from a full review.

2.2 Opinion Selection

Given the set or reviews $R_e = \{r_1, r_2, \ldots\}$ for an entity $e$, we define the entity’s opinion set as $O_e = \{O_{r_1} \cup O_{r_2} \cup \ldots\}$. Summarizing the opinions about entity $e$ relies on selecting the most salient opinions $S_e \subset O_e$. As a departure from previous work, we explicitly select the opinion phrases that will form the basis for summarization, in the following steps.

Opinion Merging: To avoid selecting redundant opinions in $S_e$, we apply a greedy algorithm to merge similar opinions into clusters $C = \{C_1, C_2, \ldots\}$: given an opinion set $O_e$, we start with an empty $C$, and iterate through every opinion in $O_e$. For each opinion, $(o_i, pol_i, a_i)$, we further iterate through every existing cluster in random order. The opinion is added to the first cluster $C$ which satisfies the following criterion, or to a newly created cluster otherwise:

$$\forall (o_j, pol_j, a_j) \in C, \ cos(v_i, v_j) \geq \theta$$

3 Summary Generation

We describe our framework in Section 2 and present two types of experiments in Section 3: A quantitative comparison against established summarization techniques on the YELP summarization corpus (Chu and Liu, 2019); and two user studies, validating the automatic results and our method’s ability for controllable summarization.
where $v_i$ and $v_j$ are the average word embedding of opinion phrase $o_i$ and $o_j$ respectively, $\cos(\cdot, \cdot)$ is the cosine similarity, and $\theta \in (0, 1]$ is a hyper-parameter. For each opinion cluster $\{C_1, C_2, \ldots\}$, we define its representative opinion $\text{Repr}(C_i)$, which is the opinion phrase closest to its centroid.

**Opinion Ranking:** We assume that larger clusters contain opinions which are popular among reviews and, therefore, should have higher priority to be included in $S_c$. We use the representative opinions of the top-$k$ largest clusters, as selected opinions $S_c$. The Opinion Merging and Ranking steps are demonstrated in Step 2 (bottom-left) of Figure 1, where the top-3 opinion clusters are shown and their representative opinions are selected.

**Opinion Filtering (optional):** We can further control the selection by filtering opinions based on their predicted aspect category or sentiment polarity. For example, we may only allow opinions where $a_i = \text{“cleanliness”}$.

### 2.3 Summary Generation

Our goal is to generate a natural language summary which articulates $S_c$, the set of selected opinions. To achieve this, we need a natural language generation (NLG) model which takes a set of opinion phrases as input and produces a fluent, review-like summary as output. Because we cannot rely on gold-standard summaries for training, we train an NLG model that encodes the extracted opinion phrases of a single review and then attempts to reconstruct the review’s full text. Then, the trained model can be used to generate summaries.

**Training via Review Reconstruction:** Having extracted $O_r$ for every review $r$ in a corpus, we construct training examples $\{T(O_r), r\}$, where $T(O_r)$ is a textualization of the review’s opinion set, where all opinion phrases are concatenated in their original order, using a special token $[\text{SEP}]$. For example:

$$O_r = \{\text{very comfy bed, clean bath}\}$$

$$T(O_r) = \text{“very comfy bed [SEP] clean bath”}$$

The $\{T(O_r), r\}$ pairs are used to train a Transformer model (Vaswani et al., 2017)\(^4\) to reconstruct review text from extracted opinions, as shown in Step 3a (top-right) of Figure 1.

\(^4\)Our framework is flexible w.r.t. the choice of the model. Using a pre-trained language model is part of future work.

### 3 Evaluation

#### 3.1 Datasets

We used two review datasets for evaluation. The public **Yelp** corpus of restaurant reviews, previously used by Chu and Liu (2019). We used a different snapshot of the data, filtered to the same specifications as the original paper, resulting in 624K training reviews. We used the same gold-standard summaries for 200 restaurants as used in Chu and Liu (2019).

We also used **Hotel**, a private hotel review dataset that consists of 688K reviews for 284 hotels collected from multiple hotel booking websites. There are no gold-standard summaries for this dataset, so systems were evaluated by humans.

#### 3.2 Baselines

**LexRank** (Erkan and Radev, 2004): A popular unsupervised extractive summarization method. It selects sentences based on centrality scores calculated on a graph-based sentence similarity.

**MeanSum** (Chu and Liu, 2019): An unsupervised multi-document abstractive summarizer that minimizes a combination of reconstruction and vector similarity losses. We only applied MeanSum to **Yelp**, due to its requirement for a pre-trained language model, which was not available for **Hotel**.

**Best Review / Worst Review** (Chu and Liu, 2019): A single review that has the highest/lowest average word overlap with the input reviews.

#### 3.3 Experimental Settings

For opinion extraction, the ABSA models are trained with 1.3K labeled review sentences for **Yelp** and 2.4K for **Hotel**. For opinion merging, we used pre-trained word embeddings

| Method          | R1   | R2   | RL   |
|-----------------|------|------|------|
| Best Review     | 27.97| 3.46 | 15.29|
| Worst Review    | 16.91| 1.66 | 11.11|
| LexRank         | 24.62| 3.03 | 14.43|
| MeanSum         | 27.86| 3.95 | 16.56|
| **OPINION**    | **29.30** | **5.77** | **18.56** |

Table 1: Summarization results on **Yelp** with ROUGE.
work outperforms all baseline approaches. Automatic Evaluation: Table 1 shows the automatic evaluation scores for our model and the baselines on YELP.

| Method         | I-score | C-score | R-score |
|----------------|---------|---------|---------|
| LexRank        | -35.4   | -32.1   | -13.5   |
| MeanSum        | 14.2    | 4.9     | 9.0     |
| OPINIONDIGEST  | 21.2    | 27.2    | 4.4     |

(a) YELP

| Method         | I-score | C-score | R-score |
|----------------|---------|---------|---------|
| LexRank        | -5.8    | -3.2    | -0.5    |
| Best Review    | -4.0    | -10.7   | 17.0    |
| OPINIONDIGEST  | 9.8     | 13.8    | -16.5   |

(b) HOTEL

Table 2: Best-Worst Scaling human evaluation.

| Method         | Fully (%) | Partially (%) | No (%) |
|----------------|-----------|---------------|--------|
| MeanSum        | 23.25 %   | 42.57 %       | 34.18 %|
| OPINIONDIGEST  | 29.77 %   | 47.91 %       | 22.32 %|

Table 3: Human evaluation results on content support.

| Method         | I-score | C-score | R-score |
|----------------|---------|---------|---------|
| LexRank        | -5.8    | -3.2    | -0.5    |
| Best Review    | -4.0    | -10.7   | 17.0    |
| OPINIONDIGEST  | 9.8     | 13.8    | -16.5   |

(b) HOTEL

Table 2: Best-Worst Scaling human evaluation.

Table 4: User study on aspect-specific summaries.

3.4 Results

Automatic Evaluation: Table 1 shows the automatic evaluation scores for our model and the baselines on YELP dataset. As shown, our framework outperforms all baseline approaches. Although OPINIONDIGEST is not a fully unsupervised framework, labeled data is only required by the opinion extractor and is easier to acquire than gold-standard summaries: on YELP dataset, the opinion extraction models are trained on a publicly available ABSA dataset (Wang et al., 2017).

Human Evaluation: We conducted three user studies to evaluate the quality of the generated summaries (more details in Appendix B).

First, we generated summaries from 3 systems (ours, LexRank and MeanSum/Best Review) for every entity in YELP’s summarization test set and 200 random entities in the HOTEL dataset, and asked judges to indicate the best and worst summary according to three criteria: informativeness (I), coherence (C), and non-redundancy (R). The systems’ scores were computed using Best-Worst Scaling (Louviere et al., 2015), with values ranging from -100 (unanimously worst) to +100 (unanimously best.) We aggregated users’ responses and present the results in Table 2(a). As shown, summaries generated by OPINIONDIGEST achieve the best informativeness and coherence scores compared to the baselines. However, OPINIONDIGEST may still generate redundant phrases in the summary.

Second, we performed a summary content support study. Judges were given 8 input reviews from YELP, and a corresponding summary produced either by MeanSum or by our system. For each summary sentence, they were asked to evaluate the extent to which its content was supported by the input reviews. Table 3 shows the proportion of summary sentences that were fully, partially, or not supported for each system. OPINIONDIGEST produced significantly more sentences with full or partial support, and fewer sentences without any support.

Finally, we evaluated our framework’s ability to generate controllable output. We produced aspect-specific summaries using our HOTEL dataset, and asked participants to judge if the summaries discussed the specified aspect exclusively, partially, or not at all. Table 4 shows that in 46.6% of the summaries exclusively summarized a specified aspect, while only 10.3% of the summaries failed to contain the aspect completely.

Example Output: Example summaries in Table 5 further demonstrate that a) OPINIONDIGEST is able to generate abstractive summaries from more than a hundred of reviews and b) produce controllable summaries by enabling opinion filtering.

The first two examples in Table 5 show summaries that are generated from 8 and 128 reviews of the same hotel. OPINIONDIGEST performs robustly even for a large number of reviews. Since our framework is not based on aggregating review representations, the quality of generated text is not affected by the number of inputs and may result in better-informed summaries. This is a significant difference to previous work (Chu and Liu, 2019;
Table 5: Example summaries on HOTEL (first two) and YELP (last four). Input opinions were filtered by the aspect categories (Asp), sentiment polarity (Pol), and # of reviews (N). Colors show the alignments between opinions and summaries. Italic denotes incorrect extraction. Underlined opinions do not explicitly appear in the summaries.

**Brainskas et al., 2019**, where averaging vectors of many reviews may hinder performance.

Finally, we provide qualitative analysis of the controllable summarization abilities of **OPINIONDIGEST**, which are enabled by input opinion filtering. As discussed in Section 2.2, we filtered input opinions based on predicted aspect categories and sentiment polarity. The examples of controlled summaries (last 4 rows of Table 5) show that **OPINIONDIGEST** can generate aspect/sentiment-specific summaries. These examples have redundant opinions and incorrect extractions in the input, but **OPINIONDIGEST** is able to convert the input opinions into natural summaries. Based on **OPINIONDIGEST**, we have built an online demo (Wang et al., 2020)\(^5\) that allows users to customize the generated summary by specifying search terms.

### 4 Conclusion

We described **OPINIONDIGEST**, a simple yet powerful framework for abstractive opinion summarization. **OPINIONDIGEST** is a combination of existing ABSA and seq2seq models and does not require any gold-standard summaries for training. Our experiments on the **YELP** dataset showed that **OPINIONDIGEST** outperforms baseline methods, including a state-of-the-art unsupervised abstractive summarization technique. Our user study and qualitative analysis confirmed that our method can generate controllable high-quality summaries, and can summarize large numbers of input reviews.

**Acknowledgements**

We thank Hayate Iso for helping debug the code. We also thank Prof. Mirella Lapata for helpful comments as well as the anonymous reviewers for their constructive feedback.

\(^5\)http://extremereader.megagon.info/
References

Reinald Kim Amplayo and Mirella Lapata. 2019. Informative and controllable opinion summarization. *arXiv preprint arXiv:1909.02322*.

Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proc. EMNLP*, pages 3675–3686.

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

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proc. ACL ’16*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.

Hongliang Dai and Yangqiu Song. 2019. Neural aspect and opinion term extraction with mined rules as weak supervision. In *Proc. ACL ’19*, pages 5268–5277.

Minqing Hu and Bing Liu. 2004a. Mining and summarizing customer reviews. In *Proc. KDD ’04*, pages 168–177.

Minqing Hu and Bing Liu. 2004b. Mining opinion features in customer reviews. In *Proc. AAAI*, volume 4, pages 755–760.

Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. 2019. Unsupervised neural single-document summarization of reviews via learning latent discourse structure and its ranking. In *Proc. ACL ’19*, pages 2142–2152.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. ACL Workshop on Text Summarization Branches Out*, pages 74–81.

Yang Liu and Mirella Lapata. 2019. Hierarchical transformers for multi-document summarization. In *Proc. ACL ’19*, pages 5070–5081.

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

Huaishao Luo, Tianrui Li, Bing Liu, and Junbo Zhang. 2019. DOER: Dual cross-shared RNN for aspect term-polarity co-extraction. In *Proc. ACL ’19*, pages 591–601.

Zhengjie Miao, Yuliang Li, Xiaolan Wang, and Wang-Chiew Tan. 2020. Snippext: Semi-supervised opinion mining with augmented data. In *Proc. WWW ’20*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proc. ICLR ’18*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get To The Point: Summarization with pointer-generator networks. In *Proc. ACL ’17*, pages 1073–1083.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. NIPS ’17*, pages 5998–6008.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proc. AAAI ’17*.

Xiaolan Wang, Yoshihiko Suhara, Natalie Nuno, Yuliang Li, Jinfeng Li, Nofar Carmeli, Stefanos Angelidis, Eser Kandogan, and Wang-Chiew Tan. 2020. ExtremeReader: An interactive explorer for customizable and explainable review summarization. In *Companion Proc. WWW ’20*, page 176180.
A Hyper-parameter Sensitivity Analysis

We present OPINIONDIGEST’s hyper-parameters and their default settings in Table 6. Among these hyper-parameters, we found that the performance of OPINIONDIGEST is relatively sensitive to the following hyper-parameters: top-\(k\) opinion (\(k\)), merging threshold (\(\theta\)), and maximum token length (\(L\)).

To better understand OPINIONDIGEST’s performance, we conducted additional sensitivity analysis of these three hyper-parameters. The results are shown in Figure 2.

**Top-\(k\) opinion vs Merging threshold:** We tested different \(k = \{10, 11, \ldots, 20, 30\}\) and \(\theta = \{0.6, 0.7, 0.8, 0.9\}\). The mean (std) of R1, R2, and RL scores were 29.2 (±0.3), 5.6 (±0.2), and 18.5 (±0.2) respectively.

**Top-\(k\) opinion vs Maximum token length:** We tested different \(k = \{10, 11, \ldots, 20, 30\}\) and \(T = \{40, 50, \ldots, 200\}\). The mean (std) of R1, R2, and RL scores were 29.2 (±0.4), 5.6 (±0.3), and 18.5 (±0.2) respectively.

The results demonstrate that OPINIONDIGEST is robust to the choice of the hyper-parameters and constantly outperforms the best-performing baseline method.

B Human Evaluation Setup

We conducted user study via crowdsourcing using the FigureEight\(^6\) platform. To ensure the quality of annotators, we used a dedicated expert-worker pool provided by FigureEight. We present the detailed setup of our user studies as follows.

**Best-Worst Scaling Task:** For each entity in the YELP and HOTEL datasets, we presented 8 input reviews and 3 automatically generated summaries to human annotators (Figure 3). The methods that generated those summaries were hidden from the annotators and the order of the summaries were shuffled for every entity. We further asked the annotators to select the best and worst summaries w.r.t. the following criteria:

- **Informativeness:** How much useful information about the business does the summary provide? You need to skim through the original reviews to answer this.
- **Coherence:** How coherent and easy to read is the summary?

**Content Support Task:** For the content support study, we presented the 8 input reviews to the annotators and an opinion summary produced from these reviews by one of the competing methods (ours or MeanSum). We asked the annotators to determine for every summary sentence, whether it is fully supported, partially supported, or not supported by the input reviews (Figure 4). We collected 3 responses per review sentence and calculated the ratio of responses for each category.

**Aspect-Specific Summary Task:** Finally, we studied the performance of OPINIONDIGEST in terms of its ability to generate controllable output.

### Table 6: List of OPINIONDIGEST hyper-parameters and the default settings.

| Hyper-parameter         | Value          |
|-------------------------|----------------|
| Word embedding          | glove.6B.300d  |
| Top-\(k\) opinion (\(k\)) | 15             |
| Merging threshold (\(\theta\)) | 0.8           |
| **Transformer model training:** |               |
| SGD learning rate       | 0.1            |
| Momentum (\(\beta\))    | 0.1            |
| Decay factor (\(\gamma\)) | 0.1         |
| Number of epochs        | 5              |
| Training batch size     | 8              |
| **Decoding algorithm:**  |               |
| Beam size               | 5              |
| Length penalty          | 0.6            |
| n-gram blocking (\(n\)) | 3              |
| Maximum token length (\(L\)) | 60          |

---

\(^6\)https://www.figure-eight.com/
We presented the summaries to human judges and asked them to judge whether the summaries discussed the specific aspect exclusively, partially, or not at all (Figure 5). We again collected 3 responses per summary and calculated the percentage of responses.
Figure 3: Screenshot of Best-Worst Scaling Task.
Restaurant Reviews:

Review 1
Wow! My order: Chicken Shawarma with a side of hummus and pita. Order of bricks. Cucumber drink. Side of garlic sauce. Side of cucumber salad. Absolutely clean filling. Taste delicious! Will you craving for more? I can't believe I hadn't heard of this restaurant sooner. After the fact I realize this place is all the rave!

Review 2
I tried to order steak kebabs but they made beef kebabs. I asked for tzatziki on the side but they covered all the meat with tzatziki. Taste is more like middle eastern. Not Mediterranean. Price is good. Taste is okay.

Review 3
Now this place is really good! I always drive past it but today I decided to stop and check it out. It is really good healthy and fresh.

Review 4
I was thinking this would be more of a sit-down restaurant where you order from the table instead of a chipotle style of Mediterranean food. Thought there would be more room inside for eating. The only thing good I had was the cucumber chilli which I would go back for. Not so much the food or service.

Review 5
Parsley Modern Mediterranean is wonderful. Very responsive staff. Food is delicious. I usually get the wraps (chicken or beef) and my go-to is Babaganoush and the warm pita bread is pretty amazing.

Review 6
Very delicious food in love with cucumber drink. I couldn't decide what I wanted and one specific gentleman whipped up something very amazing for me! By the name of Jari! Great service! Thanks you and will definitely be back!

Review 7
This is Chipotle for Mediterranean food. And it is delicious, I've only been here once because the location is very inconvenient for me and I'm extremely lazy about driving more than 5 minutes to go anywhere, but if it was closer, I'd be here all the time. It's probably better this way, I have very little self-control. If you like spicy - get the hot sauce. Mix it with the white sauce, you won't be disappointed.

Review 8
The food always taste fresh and leaves me very full without feeling tired. They have had a group of friendly people for a very long time making this place an incredible value. This is my favorite Mediterranean place.

Opinion Summary Sentences:

- I am a fan of the chicken shawarma.
  - Is this sentence supported by the above reviews?
    - Fully
    - Partially
    - No

- So good!
  - Fully
  - Partially
  - No

- I've also tried their chicken and it was delicious.
  - Fully
  - Partially
  - No

- The chicken is so tender and juicy.
  - Fully
  - Partially
  - No

- They also have a great selection of sauces and cheeses.
  - Fully
  - Partially
  - No

- Will definitely be back.
  - Fully
  - Partially
  - No

Opinion summary about the recommendation of a restaurant:

- I have been coming here a few years now and have never had a bad experience. I highly recommend the fried rice and the fried rice. They are the best I've had in a long time. I will definitely be coming back to try other things on the menu. Is the place to go to for Chinese

- Does this opinion summary discuss the restaurant's recommendation?
  - Yes, exclusively
  - Yes, partially
  - Not at all

Figure 4: Screenshot of Content Support Task.

Figure 5: Screenshot of Aspect-Specific Summary Task.