ExtremeReader: An interactive explorer for customizable and explainable review summarization

Xiaolan Wang
Megagon Labs
xiaolan@megagon.ai

Yoshihiko Suhara
Megagon Labs
yoshi@megagon.ai

Natalie Nuno
Megagon Labs
natalie@megagon.ai

Yuliang Li
Megagon Labs
yuliang@megagon.ai

Jinfeng Li
Megagon Labs
jinfeng@megagon.ai

Nofar Carmeli
Technion
snofca@cs.technion.ac.il

Stefanos Angelidis
University of Edinburgh
s.angelidis@ed.ac.uk

Eser Kandogan
Megagon Labs
eser@megagon.ai

Wang-Chiew Tan
Megagon Labs
wangchiew@megagon.ai

ABSTRACT

Building summarization systems have become a necessity due to the extensive volume and growth of online reviews. Despite extensive research on this topic, existing summarization systems generally fall short on two aspects. First, existing techniques generate static summaries which cannot be tailored to specific user needs. Second, most existing systems generate extractive summaries which selects only certain salient aspects from the summaries. Hence, they do not completely depict the overall opinion of the reviews. In this paper, we demonstrate a novel summarization system, ExtremeReader, that overcomes the limitations of existing summarization systems described above. ExtremeReader allows summaries to be tailored and explored interactively so that users can quickly find the desired information. In addition, ExtremeReader generates abstractive summaries with an underlying structure that helps users understand, explore, and seek explanations to the generated summaries.

CCS CONCEPTS

• **Information systems → Opinion summarization**.

KEYWORDS

Opinion summarization, Explanation mining

ACM Reference Format:

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 *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3366424.3383535

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7024-0/20/04.
https://doi.org/10.1145/3366424.3383535

1 INTRODUCTION

With the rise of e-commerce, reading reviews has become an integral part of the decision making procedure. A recent study[1] indicates that more than 90% of the customers read reviews before reserving an online service or purchasing a product. However, reading reviews is a tedious and time-consuming process as there can be an extensive number of reviews, many of which overlap partially in content. As a result, summarizing online reviews has received significant attention in both data mining and natural language processing communities.

Many commercial or academic summary visualization systems [3, 7] have been built for generating and exploring summaries. While these summary visualization systems are effective for reducing redundancy that occurs in reviews, there are two major limitations. First, the generated summaries are mostly static and the granularity of the summary cannot be tailored to user needs. For example, summaries that are generated by aggregating positive and/or negative sentiments over *"facility"* may be too general for users who are only interested in information about swimming pools. At the same time, it is also too detailed for users who simply wish to know the overall aggregated rating. Second, most existing summarization systems cannot explain the summary that is generated or allow users to explore different aspects of the generated summary.

To address these limitations, we develop a novel summary visualization system ExtremeReader[2] that can (1) generate both structured and abstractive textual summaries which are easier to interpret and (2) allow users to interactively explore and explain the textual/structured summary by drilling down/up on the parts of the summary to the level of desired granularity. More specifically, this demo makes the following contributions.

Customizable summary. In ExtremeReader, users can customize to focus the structured or textual summaries based on their areas of interest. The structured summary provides a concise overview of opinions mentioned in the reviews, and the textual summary allows easy interpretation.

[1] https://fanandfuel.com/no-online-customer-reviews-means-big-problems-2017/
[2] The goal of the system is to allow users to read many reviews quickly and effectively. Hence the name ExtremeReader.

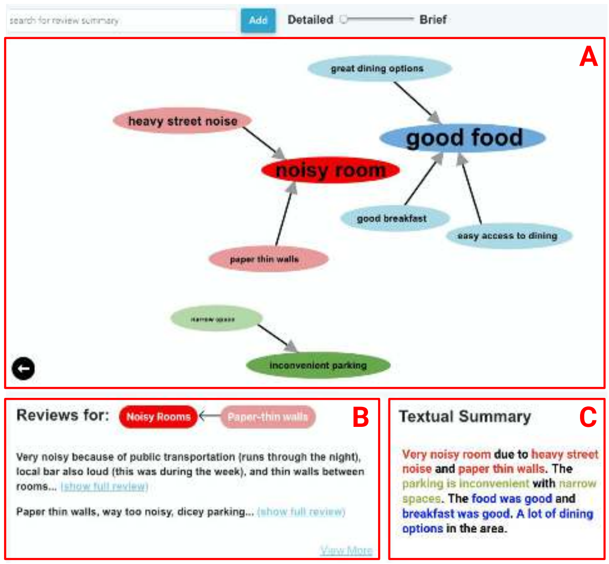

Figure 1: System overview. Our web demo has the same content but a different layout.

Explainable summary. ExtremeReader generates summaries that are explainable. Specifically, ExtremeReader explores relationship between opinions so that aspects and opinions in the generated summaries can be explained. For example, users can ask to explain *"noisy room"* in the summary and ExtremeReader will determine it is the result of *"loud sidewalk music"* based on the reviews.

Diverse interaction. Lastly, ExtremeReader is built with a set of operations for users to effectively explore the summaries. For example, users can explore the original reviews that contributed to the summaries by hovering-over our structured summaries. They can navigate finer/coarser-grained summaries through simple clicks.

The rest of this paper is organized as the follows. In Section 2 we provide an overview of ExtremeReader and describe the basic functionalities. We describe the implementation for ExtremeReader in Section 3 and conclude this paper in Section 5.

2 SYSTEM OVERVIEW

To illustrate the user experience of ExtremeReader, below we will describe a scenario.

Demonstration Scenario. John is a scholar who is planning a trip to San Francisco to attend a major academic conference. As he is going to give a keynote in the early morning, John wants to find a hotel that is *not too noisy* so that he can have a good rest. In his spare time, John plans to visit the famous Napa Valley on his rental car, so he also wants the hotel to have *easy parking*. Besides, John is interested in the recommended *dining options* as he is going to reunion with his conference friends. After browsing several hotels close to the conference location, he finds a good candidate. However, before making the final decision, he wants to make sure that this hotel satisfies all his requirements.

Unfortunately, it is not easy to find the desired information in most hotel booking websites. For example, the review summary scores *quietness*, *parking*, and *food* as *3.7*, *4.2*, and *4.8* respectively from the websites are too general to calibrate for John. Thus, John still has to read many reviews to find the right information and obtain the right understanding. In this demo, we will show how John can use ExtremeReader to quickly find the desired information by interactively customizing and explaining the the review summary.

2.1 Summary visualization

Before we describe how to use ExtremeReader, we first describe *three summary visualizations* that are provided by ExtremeReader.

Structured Summary (Figure 1 Box A). Our first visualization provides a concise structured view of the opinions mentioned in the reviews: Each node, labeled by a representative, denotes a group by semantically similar opinions. For example, "noisy room" in Figure 1 Box A represents a group of semantically similar opinions, such as "loud hotel" and "lots of noise"; Each directed edge denotes an explanation between the source node opinion (cause) and the target node opinion (outcome). For example, in Figure 1, "paper thin walls" explains "noisy room". In addition to the nodes and edges, ExtremeReader also visualizes additional information to help users gain more insights: (1). It uses different colors to highlight the *aspect category* of each node, e.g., opinions about *"food"* are color-coded with *blue*; (2). it uses different font sizes to suggest the *frequency* of the opinions: more frequently mentioned opinions are larger than less frequent ones.

Review snippet (Figure 1 Box B). Based on the structured summary, ExtremeReader further allows users to access the original reviews interactively. Essentially, for each opinion, i.e., a node in the structured summary, ExtremeReader can show the original review sentences from which the opinion was extracted. Similarly, ExtremeReader shows the original review sentences that support the opinion explanations, i.e., edges in the structured summary. For example, Figure 1 Box B shows the review snippets for an explanation: "why *paper thin walls* causes *noisy room*".

Our structured summary and review snippet visualization are supported by our structured review summarization pipeline (see Section 3.1). Through this pipeline, ExtremeReader computes a complete structured opinion graph, which includes integrated opinions, explanations, and their provenance.

Textual Summary (Figure 1 Box C). Our last visualization provides an easy-to-understand textual summary that articulates the structured summary shown in Box A. Similar to the structured summary, opinions in the textual summaries are color-coded based on the aspect information. Note that ExtremeReader always updates the textual summary alongside the structured summary.

Our textual summary is generated by a customizable abstractive summarization framework that articulates the structured summary on the fly. We present the detailed implementation of this framework in Section 3.2.

2.2 Interactive Operations

ExtremeReader is equipped with *a series of operations* that allow users to tailor and explore the summaries and reviews. In essence, John can use these operations to select a sub-graph from the precomputed structured opinion graph that matches his interest, and then explore the sub-graph interactively.

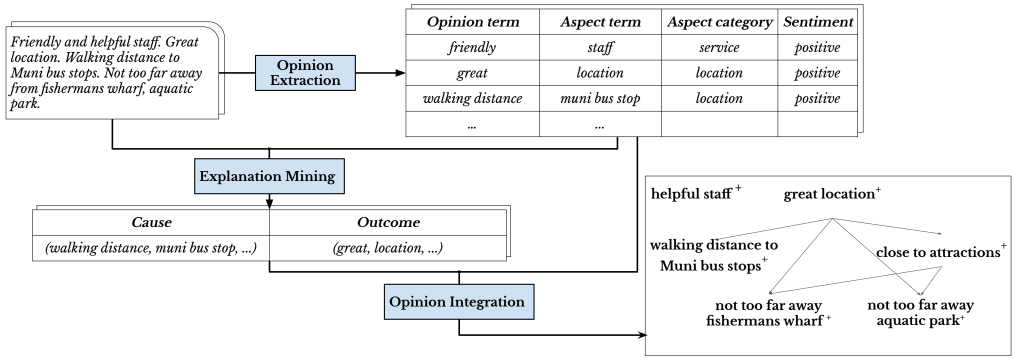

Figure 2: Structured summary generation pipeline.

Customizable Search. Through the interface shown in Figure 1, John can enter his requirements (as free text) in the search box separated by commas, e.g., "not too noisy, parking is easy, dining options". After John hits the search button, ExtremeReader will compute the sub-graph that matches with the search text he entered and display the customized structured summary in Box A. The corresponding textual summary will be displayed simultaneously in Box C. Note that entering search terms is optional. If the user chooses to not specify any search term, the system will initialize the sub-graph by the most frequent opinions. ExtremeReader also allows users to customize the number of nodes shown in the sub-graph by adjusting the sliding bar next to the search box.

Review Exploration. Users can hover-over the opinion nodes and edges in Box A to explore the original review snippets in Box B that are associated with the nodes/edges. For example, if John hovers over the edge between "noisy room" and "paper thin walls", the review snippets that support this explanation will be shown in Box B. ExtremeReader also allows users to adapt the number of review snippets shown in Box B by clicking "View More" and the full reviews can be accessed by clicking "show full review".

Drill down/up. Users can also navigate the review summaries through the drilling down/up functions: by clicking on an opinion in the structured summary (i.e., nodes in Box A), ExtremeReader will drill down and update the summary with finer-grained opinions. More specifically, from the selected opinion, ExtremeReader will reconstruct the structured summary by its top-k largest causes (ancestors in the graph). For example, when a user clicks on the opinion "noisy room", the system will summarize all finer-grained opinions that potentially explain "noisy room". Users can easily drill up by clicking the "⟵" button on the left-bottom in Box A.

With ExtremeReader, John can tailor the review summary by a simple search, explore the original reviews by hovering over the nodes and edges, or simply read through a concise and coherent textual summary. After reading the summaries, John found this hotel may not be a good fit due to the *unexpected thin walls and narrow parking, even though the dining options are great.*

3 IMPLEMENTATION

To support the functionalities, our system utilizes two core summarization tools: our first tool is a structured review summary pipeline that pre-computes the structured opinion summary; our second tool is a textual summary framework that generates controllable abstractive summary from the structured opinion (sub-)graph. In addition, we also implemented a customizable search function over the structured opinion summary, which allows users to tailor the summary more easily.

3.1 Structured review summary pipeline

Figure 2 demonstrates the structured review summary pipeline. Given a corpus of reviews, we first *extract structured opinions* from unstructured text. To make our summary explainable, we then mine the explanation relationship between opinions. Finally, we integrate semantically similar opinions and construct a concise opinion graph summary.

Opinion Extraction. Given a collection of reviews, our first goal is to extract or mine opinions from these reviews. For example, from "Friendly and helpful staff", we want to extract opinions "friendly staff" and "helpful staff". Extracting aspect and opinions is well-studied as part of the Aspect-based Sentiment Analysis (ABSA) tasks [5, 6, 8, 14]. In this demo, we use an open-source extraction pipeline [8] to extract opinions from reviews. Each extracted opinion includes an opinion term, an aspect term, an aspect category (from a list of pre-defined categories, e.g., "cleanliness", "location" in hotel domain), and the sentiment (positive/negative/neutral).

Explanation Mining. Providing explanations in summary can greatly help users gain a better understanding of the reviews. For example, customers may expect *a centrally located hotel* to be *noisy* due to the expected *heavy traffic*. However, another reason, *"paper thin walls"*, for *"noisy room"* is unexpected and may influence users' decision. To mine explanations, we define a classification task where the input includes two opinions and a review context, and the goal is to classify whether the first opinion explains the second one. Our classifier, trained with $5.9K$ crowd-sourced examples[3], achieves the state-of-the-art performance.

Opinion Integration. Finally, reviews may express similar opinions in different ways, e.g., "loud street traffic" and "heavy street noise" are essentially the same. To provide concise summaries, we want to de-duplicate opinions such that semantically similar ones appear only once in our summary. To accomplish this, we propose an unsupervised integration approach that is able to leverage various types of additional information, e.g., aspect, sentiment, and

[3]Examples are collected through https://www.figure-eight.com/.

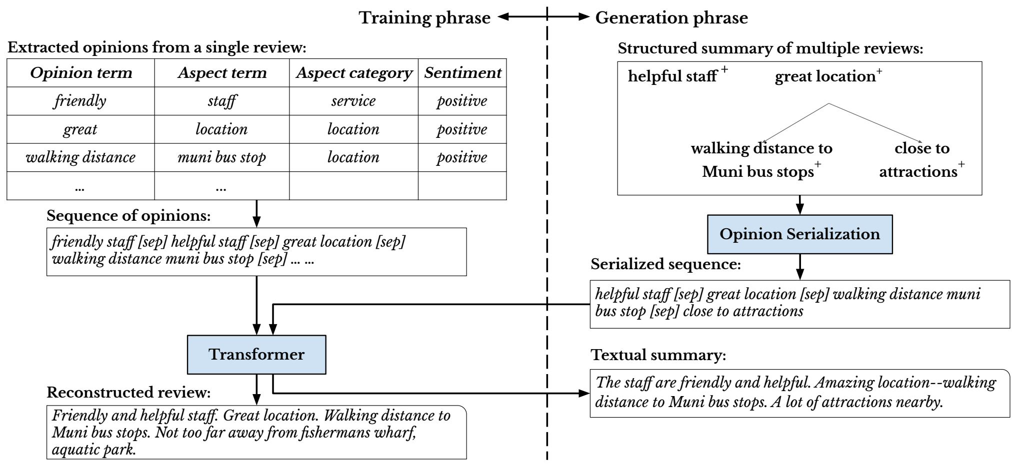

Figure 3: Textual summary generation framework.

explanations, during the integration process. Through opinion integration, we group semantically similar opinions in a cluster, treat each cluster as a node in the opinion graph, and use the centroid opinion as the representative. Based on the nodes, we refine the edges using a simple strategy: there is an edge between two clusters if such an edge exist between any of their members. Benefiting from the additional information and an embedding fine-tuning module, our proposed method achieves much higher accuracy (up to 7.9%) than baseline methods, e.g., correlation clustering.

To enable semantic search, given an opinion graph and a set of search terms, we first encode the search terms into the embedding space and then compute the similarity between the search terms and the nodes in the graph (based on the cosine similarity between their embeddings). We select the sub-graph by the most similar nodes and form our customized structured summary accordingly.

3.2 Textual review summary framework

We depict our textual review summary framework in Figure 3. Our framework uses a seq2seq model to generate a textual summary based on input opinions.

At training phrase (Figure 3 Left), our goal is to train a seq2seq model that articulates a sequence of opinions into a coherent piece of text. To accomplish this goal, we leverage the abundant resource of reviews: for each review, we first form an *opinion sequence* by concatenating its extracted opinions with a separator symbol `[SEP]`, and then train the model *to reconstruct the original review* from this opinion sequence. More specifically, we used Transformer [13], a standard seq2seq model, with the default settings for training.

At generation phrase (Figure 3 Right), given an opinion graph that is obtained from the structured review summary pipeline, we first apply *opinion serialization* on the graph to collect an *opinion sequence*. We then use our trained seq2seq model to generate the textual summary from the serialized opinion sequence. For opinion serialization, we use Breadth First Search as our serialization strategy, which makes sure that correlated opinions are close to each other in the serialized opinion sequence.

Demonstration datasets. We support two review datasets for this demonstration: a public YELP corpus of restaurant reviews (624K reviews for 9.6K restaurants) and a private dataset of hotel reviews (688K reviews for 284 hotels).

4 RELATED WORK

Early summarization efforts [5, 9, 12] mostly focused on producing (semi-)structured summaries, which show selected spans of text and statistics based on aggregated ratings. Techniques for generating such summaries include sentiment classification [11], aspect mining [4], and aspect-based sentiment analysis (ABSA) [10]. Recently, considerable research has focussed on generating unstructured textual summaries (i.e., in natural language) which are easier to interpret. Approaches to generate textual summaries are either extractive or abstractive. Extractive approaches [1] focus on selecting salient sentences in original reviews, while abstractive approaches [2] need to understand the semantics of the original reviews to produce a holistic summary of the input. The summaries produced by abstractive approaches are static, while ExtremeReader can generate customizable and explainable summaries based on the users' areas of interest.

5 CONCLUSION

In this demo, we present ExtremeReader, an interactive explorer for review summaries. To the best of our knowledge, this is the first system that is able to generate customizable and explainable summaries in both structured and abstractive textual formats. Our novel structured summary pipeline and textual summary framework also support a variety of interactive functionalities that effectively and efficiently facilitate the task of exploring and reading summaries.

REFERENCES

[1] Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proc. EMNLP '18*. 3675–3686.
[2] Eric Chu and Peter Liu. 2019. MeanSum: a neural model for unsupervised multi-document abstractive summarization. In *Proc. ICML '19*. 1223–1232.
[3] Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: A graph based approach to abstractive summarization of highly redundant opinions. In *Proc. COLING '10*. 340–348.
[4] Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *Proc. ACL '17*. 388–397.

[5] Minqing Hu and Bing Liu. 2004. Mining Opinion Features in Customer Reviews. In *Proc. AAAI '04*. 755–760.
[6] Yuliang Li, Aaron Feng, Jinfeng Li, Saran Mumick, Alon Halevy, Vivian Li, and Wang-Chiew Tan. 2019. Subjective Databases. *PVLDB*. 12, 11 (2019), 1330–1343.
[7] Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion observer: analyzing and comparing opinions on the web. In *Proc. WWW '05*. 342–351.
[8] Zhengjie Miao, Yuliang Li, Xiaolan Wang, and Wang-Chiew Tan. 2020. Snippext: Semi-supervised Opinion Mining with Augmented Data. arXiv:2002.03049 [cs.CL]
[9] Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval* 2, 1–2 (2008), 1–135.
[10] Maria Pontiki, Dimitris Galanis, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proc. SemEval '16*. 19–30.
[11] Huifeng Tang, Songbo Tan, and Xueqi Cheng. 2009. A survey on sentiment detection of reviews. *Expert Systems with Applications* 36, 7 (2009), 10760–10773.
[12] Ivan Titov and Ryan McDonald. 2008. Modeling online reviews with multi-grain topic models. In *Proc. WWW '08*. ACM, 111–120.
[13] 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*. 5998–6008.
[14] 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*.