Abstractive Summarization of Product Reviews Using Discourse Structure

Shima Gerani[‡†] [*] **Yashar Mehdad**[‡] [*] **Giuseppe Carenini**[‡] **Raymond T. Ng**[‡] **Bita Nejat**[‡]

[†]University of Lugano
Switzerland

[‡]University of British Columbia
Vancouver, BC, Canada

{gerani,mehdad,carenini,rng,nejatb}@cs.ubc.ca

Abstract

We propose a novel abstractive summarization system for product reviews by taking advantage of their discourse structure. First, we apply a discourse parser to each review and obtain a discourse tree representation for every review. We then modify the discourse trees such that every leaf node only contains the aspect words. Second, we aggregate the aspect discourse trees and generate a graph. We then select a subgraph representing the most important aspects and the rhetorical relations between them using a PageRank algorithm, and transform the selected subgraph into an aspect tree. Finally, we generate a natural language summary by applying a template-based NLG framework. Quantitative and qualitative analysis of the results, based on two user studies, show that our approach significantly outperforms extractive and abstractive baselines.

1 Introduction

Most existing works on sentiment summarization focus on predicting the overall rating on an entity (Pang et al., 2002; Pang and Lee, 2004) or estimating ratings for product features (Lu et al., 2009; Lerman et al., 2009; Snyder and Barzilay, 2007; Titov and McDonald, 2008)). However, the opinion summaries in such systems are extractive, meaning that they generate a summary by concatenating extracts that are representative of opinion on the entity or its aspects.

Comparing extractive and abstractive summaries for evaluative texts has shown that an abstractive approach is more appropriate for summarizing evaluative text (Carenini et al., 2013; Di Fabbrizio et al., 2014). This finding is also supported by a previous study in the context of summarizing news articles (Barzilay et al., 1999). To the best of our knowledge, there are only three previous works on abstractive opinion summarization (Ganesan et al., 2010; Carenini et al., 2013; Di Fabbrizio et al., 2014). The first work (Ganesan et al., 2010) proposes a graph-based method for generating ultra concise opinion summaries that are more suitable for viewing on devices with small screens. This method does not provide a well-formed grammatical abstract and the generated summary only contains words that occur in the original texts. Therefore, this approach is more extractive than abstractive. Another limitation is that the generated summaries do not contain any information about the distribution of opinions.

In the second work, (Carenini et al., 2013) addresses some of the aforementioned problems and generates well-formed grammatical abstracts that describe the distribution of opinion over the entity and its features. However, for each product, this approach requires a feature taxonomy handcrafted by humans as an input, which is not scalable. To partially address this problem (Mukherjee and Joshi, 2013) has proposed a method for the automatic generation of a product attribute hierarchy that leverages ConceptNet (Liu and Singh, 2004). However, the resulting ontology tree has been used only for sentiment classification and not for classification.

In the third and most recent study, (Di Fabbrizio et al., 2014) proposed Starlet-H as a hybrid abstractive/extractive sentiment summarizer. Starlet-H uses extractive summarization techniques to select salient quotes from the input reviews and embeds them into the abstractive summary to exemplify, justify or provide evidence for the aggregate positive or negative opinions. However, Starlet-H assumes a limited number of aspects as input and needs a large amount of training data to learn the

[*] The contribution of the first two authors to this paper was equal.

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1602–1613,
October 25-29, 2014, Doha, Qatar. ©2014 Association for Computational Linguistics

ordering of aspects for summary generation.

Highlighting the reasons behind opinions in reviews was also previously proposed in (Kim et al., 2013). However, their approach is extractive and similar to (Ganesan et al., 2010) does not cover the distribution of opinions. Furthermore, it aims to explain the opinion on only one aspect, rather than explaining the overall opinion on the product, its aspects and how they affect each other.

To address some of the above mentioned limitations , in this paper we propose a novel abstractive summarization framework that generates an aspect-based abstract from multiple reviews of a product. In our framework, anything that is evaluated in the review is considered an aspect, including the product itself. We propose a natural language generation (NLG) framework that takes aspects and their structured relation as input and generates an abstractive summary. However, unlike (Carenini et al., 2013), our method assumes no domain knowledge about the entity in terms of a user-defined feature taxonomy. On the other hand, in contrast with Starlet-H, we do not limit the input reviews to a small number of aspects and our aspect ordering method takes advantage of rhetorical information and does not require any training data. Our method relies on the discourse structure and discourse relations of reviews to infer the *importance of aspects* as well as the *association* between them (e.g., which aspects relate to each other).

Researchers have recently started using the discourse structure of text in sentiment analysis and have shown its advantage in improving sentiment classification accuracy (e.g., (Lazaridou et al., 2013; Trivedi and Eisenstein, 2013; Somasundaran et al., 2009; Asher et al., 2008)). However, to the best of our knowledge, none of the existing works have looked into exploiting discourse structure in abstractive review summarization.

In our work, *importance of aspects*, derived from the reviews' discourse structure and relations, is used to rank and select aspects to be included in the summary. More specifically, we start with the most important (highest ranked) aspects to generate a summary and add more aspects to the system until a summary of desired length is obtained. *Aspect association* is considered to better explain how the opinions on aspects affect each other (e.g., opinion over specific aspects affect the opinion over the more general ones). Consider the following sentence as an example summary generated by our system for the entity **Camera Canon G3**: "*All reviewers who commented on the camera, thought that it was really good mainly because of the photo quality.*" This summary encapsulates all the following key pieces of information: 1) *camera* and *photo quality* are the most important aspects, 2) People have positive opinion on *camera* in general and on *photo quality* as one of its features, and finally 3) *photo quality* is the main reason behind users satisfaction on *camera*. Such summary helps users understand the reason behind a rating of a product or its aspects without going through all reviews or reading scattered opinions on different aspects in multiple sentences of an extractive summary.

This paper makes the following contributions:

1. We propose a novel content selection and structuring strategy for review summarization, that assumes no prior domain knowledge, by taking advantage of the discourse structure of reviews.

2. We propose a novel product-independent template-based NLG framework to generate an abstract based on the selected content, without relying on deep syntactic knowledge or sophisticated NLG methods. Our framework, similarly to (Carenini et al., 2013), can effectively convey the distribution of opinions.

3. We present the first study that investigates the use of discourse structure information in both content selection and abstract generation for multi-document summarization.

Quantitative and qualitative analysis over evaluation results of two user studies on a set of user reviews on twelve different products show that our system is an effective abstractive system for review summarization.

2 Summarization Framework

At a high-level, our summarization framework involves generating a summary from multiple input reviews based on an Aspect Hierarchy Tree (AHT) that reflects the importance of aspects as well as the relationships between them. In our framework, an AHT is generated automatically from the set of input reviews, where each sentence of every review is marked by the aspects presented in that sentence and the polarity of opinions over them. There are various methods for extracting the aspects and predicting the polarity of opinion (Hu and Liu, 2004b; Hu and Liu, 2006; Kim et al., 2011). In this paper we do not focus on aspect extraction and sentiment prediction but rather consider the aspect and their polarity/strength (P/S) information given as input to the system. P/S scores are integer values in the range [-3, +3], where +3 is the most positive and -3 is the most negative polarity value. We also do not attempt to automatically resolve coreferences between aspects. For example, the aspect *"g3"*, *"canon g3"* and *"canon"* were manually collapsed as into *"camera"*. This preprocessing step helps to reduce the noise generated by inaccurate aspect labeling in our reviews. Figure 1 shows two sample input reviews where the aspects and their P/S scores are identified. For example, in R1, aspects *camera*, *photo quality* and *auto mode* are mentioned. The P/S values for the three aspects are [+2], [+3] and [+2] respectively which indicate positive opinion on all aspects.

The first component of our system applies a discourse parser to each review and obtains a discourse tree representation for every review (e.g. Figure 1 (a) and (b)). The discourse trees are then modified such that every leaf node only contains the aspect words. The output of the first component is an aspect-based discourse tree (ADT) for every review (e.g. Figure 1 (c) and (d)). In the second component, we aggregate the ADTs and generate a graph called Aggregated Rhetorical Relation Graph (ARRG) (e.g. Figure 1 (f)). The third component of our framework, is responsible for content selection and structuring. It takes ARRG as input, runs Weighted PageRank, and selects a subgraph (e.g. Figure 1 (g)) representing the most important aspects. Finally it transforms the selected subgraph into a tree and provides an AHT as output (e.g. Figure 1 (h)). The generated AHT is the input of the last component which generates a natural language summary by applying micro planning and sentence realization. We now describe each component of our framework in more detail.

3 Discourse Parsing

Any coherent text is structured so that we can derive and interpret the information. This structure shows how discourse units (text spans such as sentences or clauses) are connected and relate to each other. Discourse analysis aims to reveal this structure. Several theories have been proposed in the past to describe the discourse structure, among which the Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) is one of the most popular. RST divides a text into minimal atomic units, called Elementary Discourse Units (EDUs). It then forms a tree representation of a discourse called a Discourse Tree (DT) using rhetorical relations (e.g., *Elaboration*, *Explanation*, etc) as edges, and EDUs as leaves. EDUs linked by a rhetorical relation are also distinguished based on their relative importance in conveying the author's message: *nucleus* is the central part, whereas *satellite* is the peripheral part.

We use a publicly available state-of-the-art discourse parser (Joty et al., 2013)[1] to generate a DT for each product review. Figure 1 (a) and (b) show DTs for two sample reviews where dotted edges identify the satellite spans. DT1 in Figure 1 (a) shows that review R1 consists of three EDUs with two relations *Elaboration* and *Background* between them. It also shows that the first EDU (i.e. *I love camera*) is the nucleus (shown by solid line) of the relation *Elaboration* and so the rest of the document (EDUs 2 and 3) is less important and aims at elaborating on what the author meant in the first EDU. Similarly, the structure shows that the third EDU is mentioned as background information for EDU2 and so is less important for realizing the core meaning of the document.

After obtaining the DTs, we remove all words from the text spans of each EDU, except the aspect words. Thus, for each review, we have a DT where a leaf node represents the aspects occurring in the corresponding EDU. Note that there may be EDUs containing no aspects in a review. In such cases, we keep the corresponding node and mark it with no aspect. We call the resulting tree an Aspect-based Discourse Tree (ADT) which will be used in the next components. Figure 1 (c) and (d) show ADTs generated from DTs.

4 Aspect Rhetorical Relation Graph (ARRG)

In the second component, we aim at generating an ARRG for a product, based on the ADTs which are the output from the previous component. There are two motivations behind aggregating the ADTs and building the ARRG: *i*) while each ADT can be rather noisy because of the informal language of the reviews and inaccuracies from

[1] http://alt.qcri.org/discourse/Discourse_Parser_Dist.tar.gz

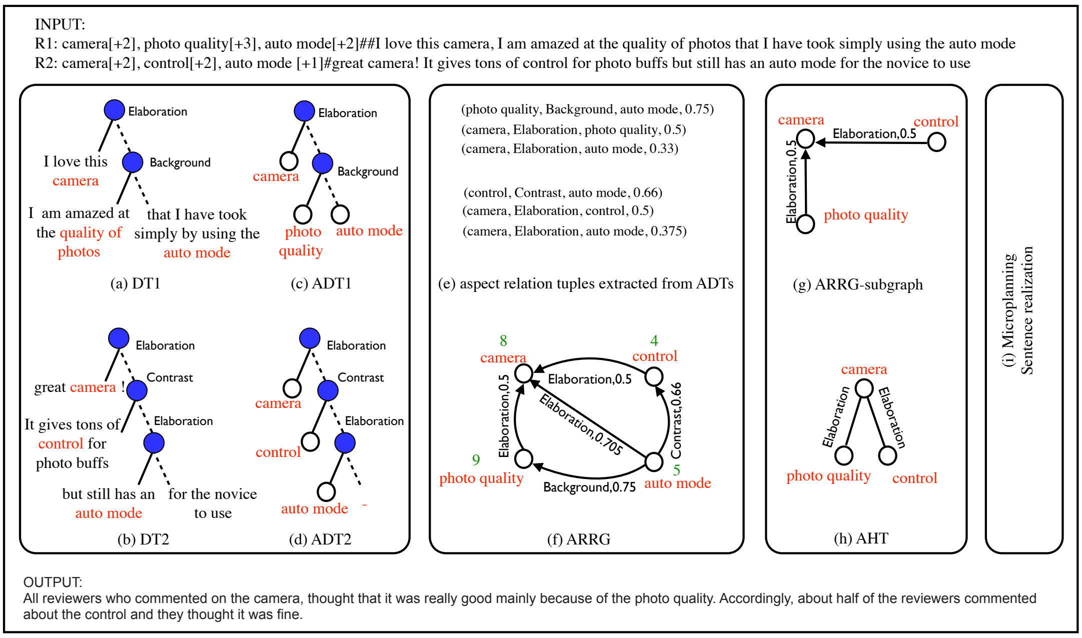
INPUT:
R1: camera[+2], photo quality[+3], auto mode[+2]##I love this camera, I am amazed at the quality of photos that I have took simply using the auto mode
R2: camera[+2], control[+2], auto mode [+1]#great camera! It gives tons of control for photo buffs but still has an auto mode for the novice to use
Elaboration
I love this camera
Background
I am amazed at the quality of photos
that I have took simply by using the auto mode
(a) DT1
Elaboration
camera
Background
photo quality
auto mode
(c) ADT1
Elaboration
great camera !
Contrast
It gives tons of control for photo buffs
Elaboration
but still has an auto mode
for the novice to use
(b) DT2
Elaboration
camera
Contrast
control
Elaboration
auto mode
(d) ADT2
(photo quality, Background, auto mode, 0.75)
(camera, Elaboration, photo quality, 0.5)
(camera, Elaboration, auto mode, 0.33)
(control, Contrast, auto mode, 0.66)
(camera, Elaboration, control, 0.5)
(camera, Elaboration, auto mode, 0.375)
(e) aspect relation tuples extracted from ADTs
8
camera
4
control
Elaboration,0.5
Elaboration,0.5
Elaboration,0.705
Contrast,0.66
9
photo quality
Background,0.75
5
auto mode
(f) ARRG
camera
control
Elaboration,0.5
Elaboration,0.5
photo quality
(g) ARRG-subgraph
camera
Elaboration
Elaboration
photo quality
control
(h) AHT
(i) Microplanning
Sentence realization
OUTPUT:
All reviewers who commented on the camera, thought that it was really good mainly because of the photo quality. Accordingly, about half of the reviewers commented about the control and they thought it was fine.

Figure 1: A simple example illustrating different components of our summarization framework.

automatic discourse parsing, aggregating all the ADTs can reveal more reliable information; and *ii*) the aggregated information highlights the most important aspects overall as well as the strongest connection between the aspects. This information can effectively drive the content selection and abstract generation phases.

ARRG is a directed graph in which we allow multiple edges between two vertices. In ARRG, vertices represent aspects. We associate to each aspect/node an importance measure that aggregates all the P/S values that the aspect receives in all the reviews. By following (Carenini et al., 2013), let $PS(a)$ be the set of P/S values that an aspect a receives. The direct measure of importance of the aspect is defined as:

$$dir\text{-}moi(a) = \sum_{ps \in PS(a)} ps^2 \qquad (1)$$

In ARRG, edges indicate existence of a rhetorical relation between text spans of a review in which the aspects occurred. Edges are labeled with the type of the relation as well as a weight indicating our confidence in the presence of the relation between the two aspects. In ARRG, an edge with label r, w from node u to node v, $u \xrightarrow{\text{r, w}} v$, indicates the existance of a relation r with confidence w between two aspects u and v. Also, the direction of the arrow indicates that u and v occurred in the *satellite* and *nucleus* spans respectively. For example, *photo quality* $\xrightarrow{\text{elaboration, 0.8}}$ *camera* indicates that there is a high confidence (0.8) that aspect *photo quality* was used in a text span to elaborate aspect *camera*. Moreover, *camera* is a more important aspect compared to *photo quality*.

To build ARRG, we use all the ADTs that are output of the previous component (one for each review). From each ADT_j, we extract all tuples of the form (u, r, v, w) in which u is an aspect occurring in a satellite span, v is an aspect occurring in a nucleus span, r is a relation type and w is the weight of the tuple computed as follows:

$$w = 1 - 0.5\frac{|\text{EDUs between u and v}|}{|\text{total EDUs in } ADT_j|} - 0.5\frac{d_r}{d} \quad (2)$$

where, $|.|$ indicates cardinality of a set. d indicates the depth of the ADT_j and d_r indicates the depth of the sub-tree of ADT_j rooted at relation r. Equation 2 weighs a tuple based on two factors: (i) the relative distance of the EDUs in which the two aspects u and v participating in relation r occur. The intuition is that aspects occurring in close proximity to each other are more related; and (ii) the depth of the sub-tree at the point of the relation relative to the depth of the whole ADT_j. This is because as we move from leaves to the root of a DT, the accuracy of the rhetorical structure has been shown to decrease. Also, at higher levels of an ADT (intra-sentential relations), it is more likely that aspects are related through non adjacent EDUs and so are less strongly related. Figure 1 (e) shows tuples extracted from sample ADTs.

Notice that every two aspects u and v may be related by the same relation more than once in an ADT for a review. Thus, we might have i tuples with the same u,r, and v but confidence weights which are not necessarily the same. From every ADT_j, we extract all (u, r, v, w_{ij}) and select the one with maximum confidence. We then aggregate the selected tuples extracted from different reviews. Putting these two steps together, for every two aspects u and v related by relation r, we obtain a single tuple $(u, r, v, \hat{w})$ where

$$\hat{w} = \sum_j \max_i w_{ij} \tag{3}$$

Figure 1 (f) shows an example ARRG built for the sample reviews.

5 Content Selection and Structuring

The content of the summary is selected by extracting from ARRG a subgraph containing the most important aspects. Such content is then structured by transforming the subgraph into an aspect hierarchy.

5.1 Subgraph Extraction

In ARRG aspects/nodes are weighted by how frequently and strongly they are evaluated in the reviews (i.e, *dir-moi*) and edges are weighted by how frequently and strongly the corresponding aspects are rhetorically related in the discourse trees (Equation 3). In content selection, we want to extract aspects that not only have high weight, but that are also linked with heavy edges to other heavy aspects. This problem can be effectively addressed by Weighted Page Rank (WPR) (Xing and Ghorbani, 2004). WPR takes the importance of both the in-links and out-links of the aspects into account and distributes rank scores based on the weights of relations between aspects. In this way, the heavier aspect nodes, that are either in the nuclei of many relations or in the satellites of relations with other heavy aspects, are promoted. We then update the weight of nodes (aspects) with the new score from WPR. Finally, we rank nodes based on their updated score *moi* and select the top *N* aspects.

$$moi(a) = \alpha dir\text{-}moi(a) + (1 - \alpha)WPR(a) \tag{4}$$

Here α is a coefficient that can be tuned on a development set or can be set to 0.5 without tuning. Figure 1 (g) shows an example subgraph selected from the sample ARRG.

5.2 Aspects Subgraph to Aspects Hierarchy Transformation

In this step, we generate a hierarchical tree structure for aspects. Such a tree structure helps to navigate over aspects and can be easily traversed to find certain aspects and their relation to their parent or children. The hierarchy of aspects also matches the intuition that the root node is the most frequent and general aspect (often the product) and as the depth increases, nodes represent more specific aspects of the product with less frequency and weight.

To obtain a hierarchical tree structure from the extracted subgraph, we first build an undirected graph as follows: we merge the edges connecting two nodes and consider the sum of their weights as the weight of the merged graph. We also ignore the relation direction for the purpose of generating the tree. We then find the Maximum Spanning Tree of the undirected subgraph and set the highest weighted aspect as the root of the tree. This process results in a useful knowledge structure of aspects with their associated weight and sentiment polarity connected with the rhetorical relations called Aspect Hierarchical Tree (AHT). Figure 1 (h) shows the generated AHT from the sub-graph.

6 Abstract Generation

The automatic generation of a natural language summary in our system involves the following tasks (Reiter and Dale, 2000): (*i*) microplanning, which covers lexical selection; and (*ii*) sentence realization, which produces english text from the output of the microplanner.

6.1 Microplanning

Once the content is selected and structured, it is passed to the microplanning module which performs lexical choice. Lexical choice is an important component of microplanning. Lexical choice is formulated in our system based on a "*formal*" style, language "*variability*" and "*fluent*" connectivity among other lexical units. Table 1 demonstrates our lexical choice strategy.

Quantifiers:
if (relative-number == 1) : [“All users (x people) who commented about the aspect”, “All costumers (x people) that reviewed the aspect”, ...]
if (relative-number >= 0.8) : [“Almost all users commented about the aspect and they”, “Almost all costumers mentioned the aspect and they”, ...]
if (relative-number >= 0.6) : [“Most users commented about the aspect and they mainly”, “Most shoppers mentioned aspect and they”, ...]
if (relative-number >= 0.45) : [“Almost half of the users commented about the aspect and they”, ’Almost 50% of the shoppers mentioned the aspect and they”, ...]
if (relative-number >= 0.2) : [“About y% of the reviewers commented about the aspect and they”, “Around y% of the shoppers mentioned the aspect and they”, ...]
if (relative-number >= 0.0) : [“z reviewers commented about the aspect and in overall they”, “z shoppers mentioned about the aspect and they”, ...]
Polarity verbs:
if (controversial(aspect)) : [“had controversial opinions about it”, “expressed controversial opinions about this feature”, ...]
else: if (average $<= -2$) : [“hated it”, “felt that it was very poor’, ’thought that it was very poor”, ...]
if (average $<= -1$) : [“disliked it”, “felt that it was poor”, “thought that it was poor”, ...]
if (average < 0) : [“did not like it”, “felt that it was weak”, “thought that it was weak”, ...]
if (average $== 0$) : [“did not express any strong positive or negative opinion about it”, ...]
if (average $<= +1$) : [“liked it”, “felt that it was fine”, “thought that it was satisfactory”, ...]
if (average $<= +2$) : [“absolutely liked it”, “really liked this feature”, “felt that it was a really good feature”, “thought that it was really good”, ...]
if (average $<= +3$) : [“loved it”, “felt that it was great”, “thought that it was great”, ...]
Connectives
[“Also, related to the aspect”, “Accordingly, ”, “Moreover, regarding the aspect, ” ,“In relation to the aspect, ”, “Talking about the aspect, ”, ...]

Table 1: Microplanning strategy for lexical choice. The selected lexical items will fill the template in the realization step.

Sentence realization templates:
First sentence templates: *if (polarity-agreement(root,highest-weighted-child) & connecting-relation == [elaboration, explain, cause, summary, same-unit, background, evidence, justify])*: *“quantifier + polarity-verb +* ‘mainly because of the’ + *highest-weighted-child”* *else: “quantifier + polarity-verb”*
First level children (aspects) sentences templates: *“connective + ‘, ’ + quantifier + ’ ’ + polarity-verb”*
Supporting sentences templates: *if (#children(aspect)==1): “connective + quantifier + verb ”* *elseif (#children(aspect)>1 & polarity-agreement(children)): “connective + quantifier + verb + [and, similarly, while, ...] + quantifier + verb”* *elseif (#children(aspect)>1 & !polarity-agreement(children)): “connective + quantifier + verb + [but, in contrast, on contrary, ...] + quantifier + verb”*

Table 2: Sentence realization templates.

Quantifiers: for each aspect, a quantifier is selected based on both the absolute and relative number of users whose opinions contributed to the evaluation of the aspect.

Polarity verbs: for each aspect, a polarity verb is selected based on the average sentiment polarity strength for that aspect. Although the average, in most cases, can be a good metric to evaluate the polarity of an aspect, it fails when the distribution of evaluations is centered on zero, for instance, if there are equal numbers of positive and negative evaluations (i.e., controversial). To partially solve this problem, we first check whether the aspect evaluation is controversial by applying the formula proposed by (Carenini and Cheung, 2008). In the case of controversiality, our microplanner selects a lexical item to express the controversiality of the aspect. In other cases, we use the average and select the polarity verb based on that.

Connectives: in order to form more fluent and readable sentences and to increase the language variability, we randomly select our connectives from the list shown in Table 1. Moreover, when a parent aspect (excluding the root in AHT) has two children, they are connected by one of the coordinating conjunction “[and, similarly]” if they agree on polarity, and they will be connected by a choice of “[on the contrary, in contrast]” otherwise (see Supporting sentences templates in Table 2). As an alternative we could have selected connectives based on the discourse relations specified in the aspects tree. However, this is left as future work.

6.2 Sentence Realization

The realization of our abstract generation is performed by applying a rather simple and comprehensive template-based strategy. Depending on the specific lexical choice in microplanning step, an appropriate template and corresponding fillers are selected as shown in Table 2. We develop three different templates: *i)* generates the first abstract sentence; *ii)* generates the abstract sentence for the aspects with no children; and *iii)* generates supporting sentences for aspects with children.

For illustration, assuming that we apply this strategy to a 5-node variation of the AHT in Figure 1 (h), where the aspect “*control*” has two children “*auto mode*” and “*setting*”, we obtain “*All reviewers (45 people) who commented on the camera, thought that it was really good mainly because of the photo quality. Accordingly, about 24% of the reviewers commented about the control and they* *thought it was fine. Also, related to the control, 7 users expressed their opinion about the auto mode and they liked it, similarly, 6 shoppers commented about the setting and they thought that it was satisfactory.*"

7 Experimental Setup

7.1 Dataset and Baselines

We conduct our experiments using the customer reviews of twelve products obtained from (Hu and Liu, 2004a): 4 digital cameras, 1 DVD player, 1 MP3 player, 2 routers, 2 phones, 1 diaper and 1 antivirus. The reviews were collected from Amazon.com and Cnet.com. We use manually annotated aspects and their associated sentiment from the same dataset.

We compare the summaries generated by our system with two state-of-the-art extractive baselines and a simpler version of our abstractive system, as follows:

1) MEAD-LexRank (LR): we use the LexRank (Erkan and Radev, 2004) implementation inside the MEAD summarization framework (Radev et al., 2004), which outperforms other algorithms implemented in the MEAD framework.

2) MEADStar (MEAD*): a state-of-the-art extractive opinion summarization system (Carenini et al., 2013), which is adapted from the open source summarization framework MEAD. MEAD* orders aspects by the number of sentences evaluating that aspect, and selects a sentence from each aspect until it reaches the word limit. The sentence that is selected for each aspect is the one with the highest sum of polarity/strength evaluations for any aspect.

3) Simple Abstractive (SA): we sort the aspects of each product based on $dir\text{-}moi$ (Equation 1). Then, for each aspect, we generate a sentence based on a simple template "*quantifier + polarity-verb*" until the summary reaches the word limit.

We limit the length of our summaries to 150 words. In our experiment we use the default parameter in Equation 4 without tuning (i.e. $\alpha = 0.5$). Our system starts the content selection process with 10 aspects and generates a summary based on a AHT with 10 aspects. We add one aspect, reproduce the AHT and regenerate the summary. We repeat this process until the word limit is reached.

7.2 Evaluation Framework

On one hand, the lack of product reviews datasets with human written summaries, and on the other hand, the difficulty of generating human-written summaries for reviews, makes review summary evaluation a very challenging task.

We evaluate the summaries generated by our system by performing two user studies based on pairwise preferences using a popular crowdsourcing service.[2] The user preference evaluation is an effective method for opinion summarization (e.g., (Lerman et al., 2009)). The main motivations behind pairwise preferences evaluation is two-fold: *i*) raters can make a preference decision more efficiently than a scoring judgment; and *ii*) rater agreement is higher in preference decisions than in scoring judgments (Ariely et al., 2003).

In both user studies, for each product, we run six pairwise comparisons for four summaries. In each rating assignment, two summaries of the same product were placed in random order. Raters were shown the name of each product along with the relevant summaries and were asked to express their preference for one summary over the other using a simple set of criteria. For two summaries S_1 and S_2 raters should choose one of the following three options: 1) Prefer S_1, 2) Prefer S_2, 3) No preference.

Raters were specifically instructed that their rating should express "*overall satisfaction with the information provided by the summary*". Raters were also asked to provide a brief comment justifying their choice. Over 48 raters participated in each study, and each comparison was evaluated by at least five raters generating more than 360 judgments for each user study. We pre-select the high skilled raters to ensure a higher quality results.

The main difference between the two user studies is that in "user study 1", we show two summaries to the raters and ask them to choose the one they prefer without showing them the original reviews. In contrast, in "user study 2", we show two summaries with links to the full text of the reviews for the raters to explore. In order to make sure that the raters read the reviews, we ask them to write a short summary of the reviews before rating the automatic summaries. We ran two different user studies because: *i*) for each product there might be many reviews to be included; *ii*) there is no guaranty that raters, in various evaluation settings, read

[2] www.crowdflower.com

System I vs System II	Agreement		No preference		Preferred Sys I		Preferred Sys II	
User Studies	1	2	1	2	1	2	1	2
LR vs MEAD*	0.33	0.75	7%	6%	35%	20%	58%	74%
LR vs SA	0.42	0.83	0%	0%	38%	21%	*62%*	*79%*
LR vs Our System	0.50	1.00	0%	3%	26%	13%	***74%***	***84%***
MEAD* vs SA	0.58	0.83	0%	0%	38%	20%	*62%*	*80%*
MEAD* vs Our System	0.67	0.50	0%	3%	25%	30%	***75%***	***67%***
SA vs Our System	0.42	0.50	12%	11%	23%	32%	**65%**	**57%**

Table 3: Results of pairwise preference user studies. Statistically significant improvements ($p < 0.01$) over the baselines are demonstrated by bold fonts. Italic fonts indicate statistical significance ($p < 0.01$) of abstractive methods (SA and Our System) over extractive approaches (LR and MEAD*).

Systems	LR		MEAD*		SA		Our System	
User Studies	1	2	1	2	1	2	1	2
Preference	33%	18%	41%	41%	49%	63%	**71%**	**69%**

Table 4: System preference results. Statistically significant improvements ($p < 0.01$) over the baselines are demonstrated by bold fonts.

the reviews (partially or completely); and *iii*) there is no evidence regarding the depth that each rater would look into the reviews. Therefore, choosing between user study 1 and 2 is not a straightforward decision. In other words, designing the two user studies in this way helps us to answer the question: "*Does the fact that raters can read all the reviews affect their ratings?*".

8 Results

This section provides a quantitative and qualitative analysis of the evaluation results[3].

8.1 Quantitative Analysis

Quantitative results for both user studies are shown in Table 3. The second column indicates the percentage of judgments for which the raters were in agreement. Agreement here is a weak agreement, where four (out of five) raters are defined to be in agreement if they all gave the same rating. The next three columns indicate the percentage of judgments for each preference category, grouped into two user studies. In addition, we measure the preference for each system in both user studies (Table 4). For each system, the preference is the number of times raters prefer the system, divided by the total number of judgments for that system (e.g., if A is preferred over B 10 out of 30 times, and A is preferred over C 15 out of 20 times, the overall preference of A is (10+15)/(30+20)=50%)

Abstractive vs. Extractive: the results of our system and SA in Table 3 show statistically significant improvements in pairwise preference over extractive baselines (LR and MEAD*) in both user studies.[4] Moreover, the results of overall preference in Table 4 demonstrates that two abstractive systems are preferred over the extractive ones in both studies. This further supports the findings in the previous studies (e.g., (Carenini et al., 2013)) that users prefer abstractive summarization. We can observe that, in both user studies, raters prefer our system over other abstractive and extractive baselines. Also, the highest pairwise preference percentages occur comparing an extractive and an abstractive system (e.g., LR vs Our System).

Abstractive Systems: the raters prefer our system over SA in both user studies (65% and 57%), and our system ranks first in our pairwise preference user studies. Knowing that both systems are abstractive and the differences between them comes from using the rhetorical structure in the content selection and abstract generation phases, proves the effectiveness of using rhetorical structure and relations in abstractive summarization of reviews.

Extractive Systems: the result in Table 3 and 4 demonstrate that raters prefer MEAD* over LR. Although both systems are extractive, the MEAD* system has been proposed for extractive opinion

[3] The evaluation results and summaries obtained from CrowdFlower are publicly available and can be downloaded from: `https://www.cs.ubc.ca/cs-research/lci/research-groups/natural-language-processing/reviews/user_study_results.zip`

[4] The statistical significance tests was calculated by approximate randomization, as described in (Yeh, 2000).

Preference Sys 1 to Sys 2	Reasons	Examples of preference justification taken from the raters comments
Our System to LR and MEAD*	Readability, coverage of aspects, aggregation of opinions	*better wording, more objective, more depth, I like the stats, more detail about people opinion, less personal experience, detail comparison from different reviews, a summary in a summary, mentions more features, ...*
LR and MEAD* to Our System	Descriptiveness, personal point of views, product capabilities	*explain how the product is positive, good characteristics about the product, has lot more to tell, more descriptive about features, personal perspective, not only characteristics but also ability, more true to the product itself, ...*
Our System to SA	The relations between the aspects, more language variability	*provides a bit more information, is very complete, not repetitive, more elegant, coherent, ...*
SA to Our System	Simpler structure, more aspects	*written better, has touched variety of features, ...*

Table 5: System preference results. The reasons are classified based on raters justifications preferring the underlined systems.

summarization. In contrast, LR is a generic extractive summarization system which is not optimized for opinion summarization. This also further demonstrates the need for opinion and reviews summarization systems.

User Study 1 vs. User Study 2: the first interesting observation is that, although the overall ranking of systems in both user studies does not change, there are some changes in the results. This indicates that reading the reviews effects preference decisions. We can observe that in all cases except one (MEAD* vs Our System) the agreement between the raters increases significantly when they are given the reviews. This can be interpreted as reading the reviews helps the rater to choose a better summary easier and more effectively. Moreover, we calculate the overall agreement for both user studies.[5] Case study 2 reports a higher overall agreement (70%) in comparison with the user study 1 (65%). This further proves our finding that showing the reviews can help the raters with their preference judgment.

In Table 3, the preference of sys 2 (last column) significantly rises for all cases when compared with the LR system. This proves that raters strongly prefer the summaries that cover opinionated sentences, specifically when they are exposed to the reviews. The same result is reflected in Table 4, where the overall preference of LR drops when the raters are given the reviews. We also observe a significant rise in preference of sys 2 when MEAD* is compared with SA (Table 3) and in the overall preference of SA (Table 4) in user study 2. This proves that raters become more confident in preferring an abstractive summary over an extractive one when the reviews are given to them. In contrast, we notice that the preference of sys 2 drops comparing "MEAD* vs Our System" and "SA vs Our System". Knowing that the drop is not significant and the the overall ranking of systems remains unchanged, this case is less straight forward to interpret.

8.2 Qualitative Analysis

We collect and group the rater justifications in the results we obtain by crowdsourcing our evaluation framework, when preferring a summary over another, in Table 5. To make the comparison more clear, Example 1 shows the summaries generated by MEAD* and our system.

Comparing our system with the extractive baselines, raters' justifications are classified in three main categories. Although the language of the extractive summaries is less formal, raters often prefer our system in terms of presentation and language. They justify their selections by expressing phrases such as "*better grammar*" or "*fewer errors*". They also comment about the coverage of aspects in the summaries generated by our system and they realize that our system was capable of aggregating the opinions for each aspect. In contrast, when they prefer the extractive summaries, they like the descriptive language of the summary and the technical details of the products that were missing in our system summaries.

We also notice that raters realize the usage of structure (AHT) in our system (both of content selection and summary generation) and they appreciate it by expressing phrases such as "*very complete*", "*more elegant*" or "*related features*". In contrast, they sometimes appreciate a simpler language in summaries generated by SA. Moreover, few raters prefer the higher coverage in SA summaries. This is mainly because not using connectives and structure in SA leaves more space to include more aspects.

[5] The agreement is calculated based on 100 randomly sampled units selected from our crowdsourcing job.

Product: Nikon Coolpix 4300

MEAD*: it is very compact but the controls are so well designed that they 're still easy to use . It 's easy for beginners to use , but has features that more serious photographers will love , so it 's an excellent camera to grow into . But overall this is a good camera with a ' really good ' picture clarity ; an exceptional close-up shooting capability .The battery life is very good , i got about 90 minutes with the lcd turned on all the time , the first time around , and i have been using it with the lcd off every now and then , and have yet needed to recharge it . Yes , the picture quality and features which are too numerous to mention are unmatched for any camera in this price range.

Our System: All reviewers (34 people), who commented on the camera, felt that it was really good mainly because of the picture. Around 26% of the reviewers expressed their opinion about the picture quality and they really liked it. Around 24% of the reviewers noted the use and they thought that it was satisfactory. Talking about the use, around 24% of the reviewers expressed their opinion about the size and they felt that it was fine. Only 6 reviewers commented about the scene mode and in overall they thought that it was satisfactory. Moreover, regarding the scene mode, 4 shoppers mentioned about the manual mode and they thought that it was satisfactory, and similarly only 4 reviewers commented about the auto mode and in overall they did not express any strong positive or negative opinion about it. Only 4 costumers mentioned the software and they felt that it was really good.

Example 1. Summaries generated by our system and MEAD* baseline for the Nikon Coolpix 4300 camera. For brevity we exclude other baselines.

9 Conclusions

We have presented a framework for abstractive summarization of product reviews based on discourse structure. For content selection, we propose a graph model based on the importance and association relations between aspects, that assumes no prior domain knowledge, by taking advantage of the discourse structure of reviews. For abstract generation, we propose a product independent template-based natural language generation (NLG) framework that takes aspects and their structured relation as input and generates an abstractive summary. Quantitative evaluation results, based on two pairwise preference user studies, show substantial improvement over extractive and abstractive baselines, including MEAD*, which is considered a state-of-the-art opinion extractive summarization system, and a simpler version of our abstractive system. In future work, we plan to extend the microplanning phase by taking advantage of the highly weighted rhetorical relations between the aspects and select connective phrases based on the discourse relations specified in the aspects tree. In addition, we plan to develop and evaluate an end-to-end system, in which the aspect extraction and polarity estimation of aspects are automated. In this way, we can achieve an end-to-end automatic summarizaion system for product reviews.

Acknowledgments

This work was supported in part by Swiss National Science Foundation (PBTIP2-145659) and NSERC Business Intelligence Network. We would like to thank the anonymous reviewers for their valuable comments. We also acknowledge Shafiq Rayhan Joty for his help regarding rhetorical parser.

References

Dan Ariely, George Loewenstein, and Drazen Prelec. 2003. "Coherent Arbitrariness": Stable Demand Curves Without Stable Preferences. *Quarterly Journal of Economics*, 118:73–105.

Nicholas Asher, Farah Benamara, and Yvette Yannick Mathieu. 2008. Distilling opinion in discourse: A preliminary study. In *Coling 2008: Companion volume: Posters and Demonstrations*, pages 5–8.

Regina Barzilay, Kathleen R. McKeown, and Michael Elhadad. 1999. Information fusion in the context of multi-document summarization. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 550–557, Stroudsburg, PA, USA. Association for Computational Linguistics.

Giuseppe Carenini and Jackie Chi Kit Cheung. 2008. Extractive vs. nlg-based abstractive summarization of evaluative text: the effect of corpus controversiality. In *INLG '08: Proceedings of the Fifth International Natural Language Generation Conference*, pages 33–41, Morristown, NJ, USA. Association for Computational Linguistics.

Giuseppe Carenini, Jackie Chi Kit Cheung, and Adam Pauls. 2013. Multi-document summarization of evaluative text. *Computational Intelligence*, 29(4):545–576.

Giuseppe Di Fabbrizio, Amanda Stent, and Robert Gaizauskas. 2014. A hybrid approach to multi-document summarization of opinions in reviews. In *Proceedings of the 8th International Natural Language Generation conference*, INLG 2014.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479, December.

Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 340–348, Stroudsburg, PA, USA. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004a. Mining and summarizing customer reviews. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2004 (KDD 2004)*, pages 168–177, Seattle, Washington.

Minqing Hu and Bing Liu. 2004b. Mining opinion features in customer reviews. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-2004)*.

Minqing Hu and Bing Liu. 2006. Opinion feature extraction using class sequential rules. In *Proceedings of AAAI 2006 Spring Sympoia on Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006)*.

Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad. 2013. Combining intra- and multi-sentential rhetorical parsing for document-level discourse analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 486–496, Sofia, Bulgaria, August. Association for Computational Linguistics.

Hyun Duk Kim, Kavita Ganesan, Parikshit Sondhi, and ChengXiang Zhai. 2011. Comprehensive review of opinion summarization.

Hyun Duk Kim, Malu Castellanos, Meichun Hsu, ChengXiang Zhai, Umeshwar Dayal, and Riddhiman Ghosh. 2013. Compact explanatory opinion summarization. In *Proceedings of the 22Nd ACM International Conference on Conference on Information &#38; Knowledge Management*, CIKM '13, pages 1697–1702, New York, NY, USA. ACM.

Angeliki Lazaridou, Ivan Titov, and Caroline Sporleder. 2013. A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1630–1639, Sofia, Bulgaria, August. Association for Computational Linguistics.

Kevin Lerman, Sasha Blair-Goldensohn, and Ryan McDonald. 2009. Sentiment summarization: Evaluating and learning user preferences. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 514–522, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hugo Liu and Push Singh. 2004. Conceptnet: A practical commonsense reasoning toolkit. *BT Technology Journal*, 22(4):211–226.

Yue Lu, ChengXiang Zhai, and Neel Sundaresan. 2009. Rated aspect summarization of short comments. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 131–140, New York, NY, USA. ACM.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Subhabrata Mukherjee and Sachindra Joshi. 2013. Sentiment aggregation using conceptnet ontology. In *Proceedings of the 6th International Joint Conference on Natural Language Processing*, IJCNLP 2013.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dragomir Radev, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Çelebi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, Jahna Otterbacher, Hong Qi, Horacio Saggion, Simone Teufel, Michael Topper, Adam Winkel, and Zhu Zhang. 2004. MEAD — A platform for multidocument multilingual text summarization. In *Conference on Language Resources and Evaluation (LREC)*, Lisbon, Portugal.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Benjamin Snyder and Regina Barzilay. 2007. Multiple aspect ranking using the good grief algorithm. In *HLT-NAACL*, pages 300–307.

Swapna Somasundaran, Galileo Namata, Janyce Wiebe, and Lise Getoor. 2009. Supervised and unsupervised methods in employing discourse relations for improving opinion polarity classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 170–179, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ivan Titov and Ryan T. McDonald. 2008. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, ACL 2008, pages 308–316. Association for Computational Linguistics.

Rakshit S. Trivedi and Jacob Eisenstein. 2013. Discourse connectors for latent subjectivity in sentiment analysis. In *HLT-NAACL*, pages 808–813. The Association for Computational Linguistics.

Wenpu Xing and Ali Ghorbani. 2004. Weighted pagerank algorithm. In *Proceedings of the Second Annual Conference on Communication Networks and Services Research*, CNSR '04, pages 305–314. IEEE Computer Society.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th conference on Computational linguistics - Volume 2*, COLING '00, pages 947–953, Stroudsburg, PA, USA. Association for Computational Linguistics.