Multi-Document Summarization of Evaluative Text

Giuseppe Carenini, Raymond Ng, and Adam Pauls
Deptartment of Computer Science
University of British Columbia Vancouver, Canada
{carenini,rng,adpauls}@cs.ubc.ca

Abstract

We present and compare two approaches to the task of summarizing evaluative arguments. The first is a sentence extraction-based approach while the second is a language generation-based approach. We evaluate these approaches in a user study and find that they quantitatively perform equally well. Qualitatively, however, we find that they perform well for different but complementary reasons. We conclude that an effective method for summarizing evaluative arguments must effectively synthesize the two approaches.

1 Introduction

Many organizations are faced with the challenge of summarizing large corpora of text data. One important application is evaluative text, i.e. any document expressing an evaluation of an entity as either positive or negative. For example, many websites collect large quantities of online customer reviews of consumer electronics. Summaries of this literature could be of great strategic value to product designers, planners and manufacturers. There are other equally important commercial applications, such as the summarization of travel logs, and non-commercial applications, such as the summarization of candidate reviews.

The general problem we consider in this paper is how to effectively summarize a large corpora of evaluative text about a single entity (e.g., a product). In contrast, most previous work on multi-document summarization has focused on factual text (e.g., news (McKeown et al., 2002), biographies (Zhou et al., 2004)). For factual documents, the goal of a summarizer is to select the most important facts and present them in a sensible ordering while avoiding repetition. Previous work has shown that this can be effectively achieved by carefully extracting and ordering the most informative sentences from the original documents in a domain-independent way. Notice however that when the source documents are assumed to contain inconsistent information (e.g., conflicting reports of a natural disaster (White et al., 2002)), a different approach is needed. The summarizer needs first to extract the information from the documents, then process such information to identify overlaps and inconsistencies between the different sources and finally produce a summary that points out and explain those inconsistencies.

A corpus of evaluative text typically contains a large number of possibly inconsistent 'facts' (i.e. opinions), as opinions on the same entity feature may be uniform or varied. Thus, summarizing a corpus of evaluative text is much more similar to summarizing conflicting reports than a consistent set of factual documents. When there are diverse opinions on the same issue, the different perspectives need to be included in the summary.

Based on this observation, we argue that any strategy to effectively summarize evaluative text about a single entity should rely on a preliminary phase of information extraction from the target corpus. In particular, the summarizer should at least know for each document: what features of the entity were evaluated, the polarity of the evaluations and their strengths.

In this paper, we explore this hypothesis by considering two alternative approaches. First, we developed a sentence-extraction based summarizer that uses the information extracted from the corpus to select and rank sentences from the corpus. We implemented this system, called MEAD*, by

adapting MEAD (Radev et al., 2003), an open-source framework for multi-document summarization. Second, we developed a summarizer that produces summaries primarily by generating language from the information extracted from the corpus. We implemented this system, called the Summarizer of Evaluative Arguments (SEA), by adapting the Generator of Evaluative Arguments (GEA) (Carenini and Moore, expected 2006) a framework for generating user tailored evaluative arguments.

We have performed an empirical formative evaluation of MEAD* and SEA in a user study. In this evaluation, we also tested the effectiveness of human generated summaries (HGS) as a topline and of summaries generated by MEAD without access to the extracted information as a baseline. The results indicate that SEA and MEAD* quantitatively perform equally well above MEAD and below HGS. Qualitatively, we find that they perform well for different but complementary reasons. While SEA appears to provide a more general overview of the source text, MEAD* seems to provide a more varied language and detail about customer opinions.

2 Information Extraction from Evaluative Text

2.1 Feature Extraction

Knowledge extraction from evaluative text about a single entity is typically decomposed into three distinct phases: the determination of features of the entity evaluated in the text, the strength of each evaluation, and the polarity of each evaluation. For instance, the information extracted from the sentence *"The menus are very easy to navigate but the user preference dialog is somewhat difficult to locate."* should be that the "menus" and the "user preference dialog" features are evaluated, and that the "menus" receive a very positive evaluation while the "user preference dialog" is evaluated rather negatively.

For these tasks, we adopt the approach described in detail in (Carenini et al., 2005). This approach relies on the work of (Hu and Liu, 2004a) for the tasks of strength and polarity determination. For the task of feature extraction, it enhances earlier work (Hu and Liu, 2004c) by mapping the extracted features into a hierarchy of features which describes the entity of interest. The resulting mapping reduces redundancy and provides conceptual organization of the extracted features.

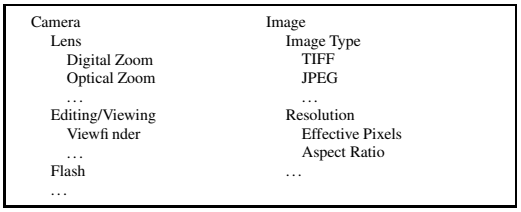

Figure 1: Partial view of UDF taxonomies for a digital camera.

Before continuing, we shall describe the terminology we use when discussing the extracted knowledge. The features evaluated in a corpus of reviews and extracted by following Hu and Liu's approach are called Crude Features.

$$CF = \{cf_j\} \;\; j = 1...n$$

For example, crude features for a digital camera might include "picture quality", "viewfinder", and "lens". Each sentence s_k in the corpus contains a set of evaluations (of crude features) called $eval(s_k)$. Each evaluation contains both a polarity and a strength represented as an integer in the range $[-3,-2,-1,+1,+2,+3]$ where $+3$ is the most positive possible evaluation and -3 is the most negative possible evaluation.

There is also a hierarchical set of possibly more abstract user-defined features [1]

$$UDF = \{udf_i\} \;\; i = 1...m$$

See Figure 1 for a sample UDF. The process of hierarchically organizing the extracted features produces a mapping from CF to UDF features (see (Carenini et al., 2005) for details). We call the set of crude features mapped to the user-defined feature udf_i $map(udf_i)$. For example, the crude features "unresponsiveness", "delay", and "lag time" would all be mapped to the udf "delay between shots".

For each cf_j, there is a set of polarity and strength evaluations $ps(cf_j)$ corresponding to each evaluation of cf_j in the corpus. We call the set of polarity/strength evaluations directly associated with udf_i

$$PS_i = \bigcup_{cf_j \varepsilon map(udf_i)} ps(cf_j)$$

The total set of polarity/strength evaluations associated with udf_i, including its descendants, is

[1] We call them here user-defi ned features for consistency with (Carenini et al., 2005). In this paper, they are not assumed to be and are not in practice defi ned by the user.

$$TPS_i = PS_i \bigcup \left\{ \bigcup_{udf_k \varepsilon desc(udf_i)} PS_k \right\}$$

where $desc(udf_i)$ refers to all descendants of udf_i.

3 MEAD*: Sentence Extraction

Most modern summarization systems use sentences extracted from the source text as the basis for summarization (see (Nat, 2005b) for a representative sample). Extraction-based approaches have the advantage of avoiding the difficult task of natural language generation, thus maintaining domain-independence because the system need not be aware of specialized vocabulary for its target domain. The main disadvantage of extraction-based approaches is the poor linguistic coherence of the extracted summaries.

Because of the widespread and well-developed use of sentence extractors in summarization, we chose to develop our own sentence extractor as a first attempt at summarizing evaluative arguments. To do this, we adapted MEAD (Radev et al., 2003), an open-source framework for multi-document summarization, to suit our purposes. We refer to our adapted version of MEAD as MEAD*. The MEAD framework decomposes sentence extraction into three steps: (i) *Feature Calculation*: Some numerical feature(s) are calculated for each sentence, for example, a score based on document position and a score based on the TF*IDF of a sentence. (ii) *Classification*: The features calculated during step (i) are combined into a single numerical score for each sentence. (iii) *Reranking*: The numerical score for each sentence is adjusted relative to other sentences. This allows the system to avoid redundancy in the final set of sentences by lowering the score of sentences which are similar to already selected sentences.

We found from early experimentation that the most informative sentences could be accurately determined by examining the extracted CFs. Thus, we created our own sentence-level feature based on the number, strength, and polarity of CFs extracted for each sentence.

$$CF\_sum(s_k) = \sum_{ps_i \varepsilon\ eval(s_k)} |ps_i|$$

During system development, we found this measure to be effective because it was sensitive to the number of CFs mentioned in a given sentence as well as to the strength of the evaluation for each CF. However, many sentences may have the same *CF_sum* score (especially sentences which contain an evaluation for only one CF). In such cases, we used the MEAD 3.07[2] centroid feature as a 'tie-breaker'. The centroid is a common feature in multidocument summarization (cf. (Radev et al., 2003), (Saggion and Gaizauskas, 2004)).

At the reranking stage, we adopted a different algorithm than the default in MEAD. We placed each sentence which contained an evaluation of a given CF into a 'bucket' for that CF. Because a sentence could contain more than one CF, a sentence could be placed in multiple buckets. We then selected the top-ranked sentence from each bucket, starting with the bucket containing the most sentences (largest $|ps(cf_j)|$), never selecting the same sentence twice. Once one sentence had been selected from each bucket, the process was repeated[3]. This selection algorithm accomplishes two important tasks: firstly, it avoids redundancy by only selecting one sentence to represent each CF (unless all other CFs have already been represented), and secondly, it gives priority to CFs which are mentioned more frequently in the text.

The sentence selection algorithm permits us to select an arbitrary number of sentences to fit a desired word length. We then ordered the sentences according to a primitive discourse planning strategy in which the most general CF (i.e. the CF mapped to the topmost node in the UDF) is discussed first. The remaining sentences were then ordered according to a depth-first traversal of the UDF hierarchy. In this way, general features are followed immediately by their more specific children in the hierarchy.

4 SEA: Natural Language Generation

The extraction-based approach described in the previous section has several disadvantages. We already discussed problems with the linguistic coherence of the summary, but more specific problems arise in our particular task of summarizing a corpus of evaluative text. Firstly, sentence extraction does not give the reader any explicit information about of the distribution of evaluations, for example, how many users mentioned a given fea-

[2]The centroid calculation requires an IDF database. We constructed an IDF database from several corpora of reviews and a set of stop words.

[3]In practice the process would only be repeated in summaries long enough to contain sentences for each CF, which is very rare.

ture and whether user opinions were uniform or varied. It also does not give an aggregate view of user evaluations because typically it only presents one evaluation for each CF. It may be that a very positive evaluation for one CF was selected for extraction, even though most evaluations were only somewhat positive and some were even negative.

We thus also developed a system, SEA, that presents such information in generated natural language. This system calculates several important characteristics of the source corpus by aggregating the extracted information including the CF to UDF mapping. We first describe these characteristics and then discuss their presentation in natural language.

4.1 Aggregation of Extracted Information

In order to provide an aggregate view of the evaluation expressed in a corpus of evaluative text a summarizer should at least determine: (i) which features of the evaluated entity were most 'important' to the users (ii) some aggregate of the user opinions for the important features (iii) the distribution of those opinions and (iv) the reasons behind each user opinion. We now discuss each of these aspects in detail.

4.1.1 Feature Selection

We approach the task of selecting the most 'important' features by defining a 'measure of importance' for each feature of the evaluated entity. We define the 'direct importance' of a feature in the UDF as

$$dir\_moi(udf_i) = \sum_{ps_k \varepsilon PS_i} |ps_k|^2$$

where by 'direct' we mean the importance derived only from that feature and not from its children. This metric produces high scores for features which either occur frequently in the corpus or have strong evaluations (or both). This 'direct' measure of importance, however, is incomplete, as each non-leaf node in the UDF effectively serves a dual purpose. It is both a feature upon which a user might comment and a category for grouping its sub-features. Thus, a non-leaf node should be important if either its children are important or the node itself is important (or both). To this end, we have defined the total measure of importance $moi(udf_i)$ as

$$\begin{cases} dir\_moi(udf_i) & ch(udf_i) = \emptyset \\ [\alpha\, dir\_moi(udf_i) + & \\ (1-\alpha) \sum_{udf_k \varepsilon ch(udf_i)} moi(udf_k)] & \text{otherwise} \end{cases}$$

where $ch(udf_i)$ refers to the children of udf_i in the hierarchy and α is some real parameter in the range $[0.5, 1]$. In this measure, the importance of a node is a combination of its direct importance and of the importance of its children. The parameter α may be adjusted to vary the relative weight of the parent and children. We used $\alpha = 0.9$ for our experiments. This setting resulted in more informative summaries during system development.

In order to perform feature selection using this metric, we must also define a selection procedure. The most obvious is a simple greedy selection – sort the nodes in the UDF by the measure of importance and select the most important node until a desired number of features is included. However, because a node derives part of its 'importance' from its children, it is possible for a node's importance to be dominated by one or more of its children. Including both the child and parent node would be redundant because most of the information is contained in the child. We thus choose a dynamic greedy selection algorithm in which we recalculate the importance of each node after each round of selection, with all previously selected nodes removed from the tree. In this way, if a node that dominates its parent's importance is selected, its parent's importance will be reduced during later rounds of selection. This approach mimics the behaviour of several sentence extraction-based summarizers (e.g. (Schiffman et al., 2002; Saggion and Gaizauskas, 2004)) which define a metric for sentence importance and then greedily select the sentence which minimizes similarity with already selected sentences and maximizes informativeness.

4.1.2 Opinion Aggregation

We approach the task of aggregating opinions from the source text in a similar fashion to determining the measure of importance. We calculate an 'orientation' for each UDF by aggregating the polarity/strength evaluations of all related CFs into a single value. We define the 'direct orientation' of a UDF as the average of the strength/polarity evaluations of all related CFs

$$dir\_ori(udf_i) = \underset{ps_k \varepsilon PS_i}{avg}\ ps_k$$

As with our measure of importance, we must also include the orientation of a feature's children in its orientation. Because a feature in the UDF conceptually groups its children, the orientation of a feature should include some information about the orientation of its children. We thus define the total orientation $ori(udf_i)$ as

$$\begin{cases} dir\_ori(udf_i) & ch(udf_i) = \emptyset \\ [\alpha\, dir\_ori(udf_i) + & \\ (1-\alpha)\ avg_{udf_k \varepsilon ch(udf_i)}\, ori(udf_k)] & \text{otherwise} \end{cases}$$

This metric produces a real number between -3 and $+3$ which serves as an aggregate of user opinions for a feature. We use the same value of α as in $moi(udf_i)$.

4.1.3 Distribution of Opinions

Communicating user opinions to the reader is not simply a matter of classifying each feature as being evaluated negatively or positively – the reader may also want to know if all users evaluated a feature in a similar way or if evaluations were varied. We thus also need a method of determining the modality of the distribution of user opinions. We calculate the sum of positive polarity/strength evaluations (or negative if $ori(udf_i)$ is negative) for a node and its children as a fraction of all polarity/strength evaluations

$$\frac{\sum_{v_i \varepsilon \{ps_k \varepsilon TPS_i | signum(ps_k) = signum(ori(udf_i))\}} |v_i|}{\sum_{v_i \varepsilon TPS_i} |v_i|}$$

If this fraction is very close to 0.5, this indicates an almost perfect split of user opinions on that features. So we classify the feature as 'bimodal' and we report this fact to the user. Otherwise, the feature is classified as 'unimodal', i.e. we need only to communicate one aggregate opinion to the reader.

4.2 Generating Language: Adapting the Generator of Evaluative Arguments (GEA)

The first task in generating a natural language summary from the information extracted from the corpus is content selection. This task is accomplished in SEA by the feature selection strategy described in Section 4.1.1. After content selection, the automatic generation of a natural language summary involves the following additional tasks (Reiter and Dale, 2000): (i) structuring the content by ordering and grouping the selected content elements as well as by specifying discourse relations (e.g., supporting vs. opposing evidence) between the resulting groups; (ii) microplanning, which involves lexical selection and sentence planning; and (iii) sentence realization, which produces English text from the output of the microplanner. For most of these tasks, we have adapted the Generator of Evaluative Arguments (GEA) (Carenini and Moore, expected 2006), a framework for generating user tailored evaluative arguments. For lack of space we cannot discuss the details here. These are provided on the online version of this paper, which is available at the first author's Web page. That version also includes a detailed discussion of related and future work.

5 Evaluation

We evaluated our two summarizers by performing a user study in which four treatments were considered: SEA, MEAD*, human-written summaries as a topline and summaries generated by MEAD (with all options set to default) as a baseline.

5.1 The Experiment

Twenty-eight undergraduate students participated in our experiment, seven for each treatment. Each participant was given a set of 20 customer reviews randomly selected from a corpus of reviews. In each treatment three participants received reviews from a corpus of 46 reviews of the Canon G3 digital camera and four received them from a corpus of 101 reviews of the Apex 2600 Progressive Scan DVD player, both obtained from Hu and Liu (2004b). The reviews from these corpora which serve as input to our systems have been manually annotated with crude features, strength, and polarity. We used this 'gold standard' for crude feature, strength, and polarity extraction because we wanted our experiments to focus on our summary and not be confounded by errors in the knowledge extraction phase.

The participant was told to pretend that they work for the manufacturer of the product (either Canon or Apex). They were told that they would have to provide a 100 word summary of the reviews to the quality assurance department. The purpose of these instructions was to prime the user to the task of looking for information worthy of summarization. They were then given 20 minutes to explore the set of reviews.

After 20 minutes, the participant was asked to stop. The participant was then given a set of in-

structions which explained that the company was testing a computer-based system for automatically generating a summary of the reviews s/he has been reading. S/he was then shown a 100 word summary of the 20 reviews generated either by MEAD, MEAD*, SEA, or written by a human [4]. Figure 2 shows four summaries of the same 20 reviews, one of each type.

In order to facilitate their analysis, summaries were displayed in a web browser. The upper portion of the browser contained the text of the summary with 'footnotes' linking to reviews on which the summary was based. For MEAD and MEAD*, for each sentence the footnote pointed to the review from which the sentence had been extracted. For SEA and human-generated summaries, for each aggregate evaluation the footnote pointed to the review containing a sample sentence on which that evaluation was based. In all summaries, clicking on one of the footnotes caused the corresponding review to be displayed in which the appropriate sentence was highlighted.

Once finished, the participant was asked to fill out a questionnaire assessing the summary along several dimensions related to its effectiveness. The participant could still access the summary while s/he worked on the questionnaire.

Our questionnaire consisted of nine questions. The first five questions were the SEE linguistic well-formedness questions used at the 2005 Document Understanding Conference (DUC) (Nat, 2005a). The next three questions were designed to assess the content of the summary. We based our questions on the Responsive evaluation at DUC 2005; however, we were interested in a more specific evaluation of the content that one overall rank. As such, we split the content into the following three separate questions:

- (Recall) *The summary contains all of the information you would have included from the source text.*
- (Precision) *The summary contains no information you would NOT have included from the source text.*
- (Accuracy) *All information expressed in the summary accurately reflects the information contained in the source text.*

The final question in the questionnaire asked the participant to rank the overall quality of the summary holistically.

[4]For automatically generated summaries, we generated the longest possible summary with less than 100 words.

5.2 Quantitative Results

Table 1 consists of two parts. The first top half focuses on linguistic questions while the second bottom half focuses on content issues. We performed a two-way ANOVA test with summary type as rows and the question sets as columns. Overall, it is easy to conclude that MEAD* and SEA performed at a roughly equal level, while the baseline MEAD performed significantly lower and the Human summarizer significantly higher ($p < .001$). When individual questions/categories are considered, there are few questions that differentiate between MEAD* and SEA with a p-value below 0.05. The primary reason is our small sample size. Nonetheless, if we relax the p-value threshold, we can make the following observations/hypotheses. To validate some of these hypotheses, we would conduct a larger user study in future work.

On the linguistic side, the average score suggests the ordering of: $Human > \{MEAD*, SEA\} > MEAD$. Both MEAD* and SEA are also on par with the median DUC score (Nat, 2005b). On the focus question, in fact, SEA's score is tied with the Human's score, which may be a beneficial effect of the UDF guiding content structuring in a top-down fashion. It is also interesting to see that SEA outperforms MEAD* on grammaticality, showing that the generative text approach may be more effective than simply extracting sentences on this aspect of grammaticality. On the other hand, MEAD* outperforms SEA on non-redundancy, and structure and coherence. SEA's disappointing performance on structure and coherence was among the most surprising finding. One possibility is that our adaptation of GEA content structuring strategy was suboptimal or even inappropriate. We plan to investigate possible causes in the future.

On the content side, the average score suggests the ordering of: $Human > SEA > MEAD* > MEAD$. As for the three individual content questions, on the recall one, both SEA and MEAD* were dominated by the Human summarizer. This indicates that both SEA and MEAD* omit some features considered important. We feel that if a longer summary was allowed, the gap between the two and the Human summarizer would be narrower. The precision question is somewhat surprising in that SEA actually performs better than the Human summarizer. In general this indicates that the feature selection strategy was quite suc-

MEAD*: Bottom line , well made camera , easy to use , very flexible and powerful features to include the ability to use external flash and lense / filters choices . 1It has a beautiful design , lots of features , very easy to use , very configurable and customizable , and the battery duration is amazing ! Great colors , pictures and white balance. The camera is a dream to operate in automode , but also gives tremendous flexibility in aperture priority , shutter priority , and manual modes . I 'd highly recommend this camera for anyone who is looking for excellent quality pictures and a combination of ease of use and the flexibility to get advanced with many options to adjust if you like.
SEA: Almost all users loved the Canon G3 possibly because some users thought the physical appearance was very good. Furthermore, several users found the manual features and the special features to be very good. Also, some users liked the convenience because some users thought the battery was excellent. Finally, some users found the editing/viewing interface to be good despite the fact that several customers really disliked the viewfinder . However, there were some negative evaluations. Some customers thought the lens was poor even though some customers found the optical zoom capability to be excellent. Most customers thought the quality of the images was very good.
MEAD: I am a software engineer and am very keen into technical details of everything i buy , i spend around 3 months before buying the digital camera ; and i must say , g3 worth every single cent i spent on it . I do n't write many reviews but i 'm compelled to do so with this camera . I spent a lot of time comparing different cameras , and i realized that there is not such thing as the best digital camera . I bought my canon g3 about a month ago and i have to say i am very satisfied .
Human: The Canon G3 was received exceedingly well. Consumer reviews from novice photographers to semi-professional all listed an impressive number of attributes, they claim makes this camera superior in the market. Customers are pleased with the many features the camera offers, and state that the camera is easy to use and universally accessible. Picture quality, long lasting battery life, size and style were all highlighted in glowing reviews. One flaw in the camera frequently mentioned was the lens which partially obsructs the view through the view finder, however most claimed it was only a minor annoyance since they used the LCD sceen.

Figure 2: Sample automatically generated summaries.

	SEA		**MEAD***		MEAD		Human		DUC		
Question	Avg.	Dev.	Avg.	Dev.	Avg.	Dev.	Avg.	Dev.	Med.	Min.	Max.
Grammaticality	3.43	1.13	2.71	0.76	3.14	0.90	4.29	0.76	3.86	2.60	4.34
Non-redundancy	3.14	1.57	3.86	0.90	3.57	0.98	4.43	1.13	4.44	3.96	4.74
Referential clarity	3.86	0.69	4.00	1.15	3.00	1.15	4.71	0.49	2.98	2.16	4.14
Focus	4.14	0.69	3.71	1.60	2.29	1.60	4.14	0.69	3.16	2.38	3.94
Structure and Coherence	2.29	0.95	3.00	1.41	1.86	0.90	4.43	0.53	2.10	1.60	3.24
Linguistic Average	*3.37*	*1.19*	*3.46*	*1.24*	*2.77*	*1.24*	*4.4*	*0.74*	*3.31*	*2.54*	*4.08*
Recall	2.33	1.03	2.57	0.98	1.57	0.53	3.57	1.27	–	–	–
Precision	4.17	1.17	3.50	1.38	2.17	1.17	3.86	1.07	–	–	–
Accuracy	4.00	0.82	3.57	1.13	2.57	1.4	4.29	0.76	–	–	–
Content Average	*3.5*	*1.26*	*3.21*	*1.2*	*2.1*	*1.12*	*3.9*	*1.04*	–	–	–
Overall	3.14	0.69	3.14	1.21	2.14	1.21	4.43	0.79	–	–	–
Macro Average	*3.39*	*0.73*	*3.34*	*0.51*	*2.48*	*0.65*	*4.24*	*0.34*	–	–	–

Table 1: Quantative results of user responses to our questionnaire on a scale from 1 (Strongly Disagree) to 5 (Strongly Agree).

cessful. Finally, for the accuracy question, SEA is closer to the Human summarizer than MEAD*. In sum, recall that for evaluative text, it is very possible that different reviews express different opinions on the same question. Thus, for the summarization of evaluative text, when there is a difference in opinions, it is desirable that the summary accurately covers both angles or conveys the disagreement. On this count, according to the scores on the precision and accuracy questions, SEA appears to outperform MEAD*.

5.3 Qualitative Results

*MEAD**: The most interesting aspect of the comments made by participants who evaluated MEAD*-based summaries was that they rarely criticized the summary for being nothing more than a set of extracted sentences. For example, one user claimed that the summary had a "*simple sentence first, then ideas are fleshed out, and ends with a fun impact statement*". Other users, while noticing that the summary was solely quotation, still felt the summary was adequate ("*Shouldn't just copy consumers . . . However, it summarized various aspects of the consumer's opinions . . .* ").

With regard to content, two main complaints by participants were: (i) the summary did not reflect overall opinions (e.g., included positive evaluations of the DVD player even though most evaluations were negative), and (ii) the evaluations of some features were repeated. The first complaint is consistent with the relatively low score of MEAD* on the accuracy question.

We could address this complaint by only including sentences whose *CF* evaluations have polarities matching the majority polarity for each *CF*. The second complaint could be avoided by not selecting sentences which contain evaluations of *CFs* already in the summary.

SEA: Comments about the structure of the summaries generated by SEA mentioned the "coherent but robotic" feel of the summaries, the repetition of "users/customers" and lack of pronoun use, the lack of flow between sentences, and the repeated use of generic terms such as "good". These problems are largely a result of simplistic microplanning and seems to contradict SEA's disappointing performance on the structure and coherence ques-

tion.

In terms of content, there were two main sets of complaints. Firstly, participants wanted more "details" in the summary, for instance, they wanted examples of the "manual features" mentioned by SEA. Note that this is one complaint absent from the MEAD* summaries. That is, where the MEAD* summaries lack structure but contain detail, SEA summaries provide a general, structured overview while lacking in specifics.

The other set of complaints related to the problem that participants disagreed with the choice of features in the summary. We note that this is actually a problem common to MEAD* and even the Human summarizer. The best example to illustrate this point is on the "physical appearance" of the digital camera. One reason participants may have disagreed with the summarizer's decision to include the physical appearance in the summary is that some evaluations of the physical appearance were quite subtle. For example, the sentence "*This camera has a design flaw*" was annotated in our corpus as evaluating the physical appearance, although not all readers would agree with that annotation.

6 Conclusions

We have presented and compared a sentence extraction- and language generation based approach to summarizing evaluative text. A formative user study of our MEAD* and SEA summarizers found that, quantitatively, they performed equally well relative to each other, while significantly outperforming a baseline standard approach to multidocument summarization. Trends that we identified in the results as well as qualitative comments from participants in the user study indicate that the summarizers have different strengths and weaknesses. On the one hand, though providing varied language and detail about customer opinions, MEAD* summaries lack in accuracy and precision, failing to give and overview of the opinions expressed in the evaluative text. On the other, SEA summaries provide a general overview of the source text, while sounding 'robotic', repetitive, and surprisingly rather incoherent.

Some of these differences are, fortunately, quite complimentary. We plan in the future to investigate how SEA and MEAD* can be integrated and improved.

References

G. Carenini and J. D. Moore. expected 2006. Generating and evaluating evaluative arguments. *AI Journal (accepted for publication, contact fi rst author for a draft)*.

G. Carenini, R.T Ng, and E. Zwart. 2005. Extracting knowlede from evaluative text. In *Proc. Third International Conference on Knowledge Capture*.

M. Hu and B. Liu. 2004a. Mining and summarizing customer reviews. In *Proc. of the 10th ACM SIGKDD Conf.*, pages 168–177, New York, NY, USA. ACM Press.

Minqing Hu and Bing Liu. 2004b. Feature based summary of customer reviews dataset. *http://www.cs.uic.edu/ liub/FBS/FBS.html*.

Minqing Hu and Bing Liu. 2004c. Mining opinion features in customer reviews. In *Proc. AAAI*.

K. R. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, J. L. Klavans, A. Nenkova, C. Sable, B. Schiffman, and S. Sigelman. 2002. Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proceedings of the HLT Conf.*

2005a. Linguistic quality questions from the 2005 DUC. *http://duc.nist.gov/duc2005/quality-questions.txt*.

2005b. Proc. of DUC 2005.

D. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. elebi, D. Liu, and E. Drabek. 2003. Evaluation challenges in large-scale document summarization. In *Proc. of the 41st ACL*, pages 375–382.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Studies in Natural Language Processing. Cambridge University Press.

H. Saggion and R. Gaizauskas. 2004. Multi-document summarization by cluster/profi le relevance and redundancy removal. In *Proc. of DUC04*.

B. Schiffman, A. Nenkova, and K. McKeown. 2002. Experiments in multidocument summarization. In *Proc. of HLT02*, San Diego, Ca.

M. White, C. Cardie, and V. Ng. 2002. Detecting discrepancies in numeric estimates using multidocument hypertext summaries. In *Proc of HLT02*.

L. Zhou, M. Ticrea, and E. Hovy. 2004. Multi-document biography summarization. In *Proceedings of EMNLP*.