

Optimizing Informativeness and Readability for Sentiment Summarization

Hitoshi Nishikawa, Takaaki Hasegawa, Yoshihiro Matsuo and Genichiro Kikui

NTT Cyber Space Laboratories, NTT Corporation

1-1 Hikari-no-oka, Yokosuka, Kanagawa, 239-0847 Japan

{ nishikawa.hitoshi, hasegawa.takaaki }
{ matsuo.yoshihiro, kikui.genichiro } @lab.ntt.co.jp

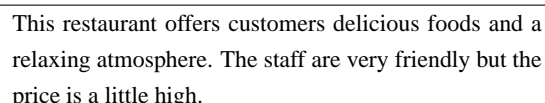
Abstract

We propose a novel algorithm for sentiment summarization that takes account of informativeness and readability, simultaneously. Our algorithm generates a summary by selecting and ordering sentences taken from multiple review texts according to two scores that represent the informativeness and readability of the sentence order. The informativeness score is defined by the number of sentiment expressions and the readability score is learned from the target corpus. We evaluate our method by summarizing reviews on restaurants. Our method outperforms an existing algorithm as indicated by its ROUGE score and human readability experiments.

1 Introduction

The Web holds a massive number of reviews describing the sentiments of customers about products and services. These reviews can help the user reach purchasing decisions and guide companies' business activities such as product improvements. It is, however, almost impossible to read all reviews given their sheer number.

These reviews are best utilized by the development of automatic text summarization, particularly sentiment summarization. It enables us to efficiently grasp the key bits of information. Sentiment summarizers are divided into two categories in terms of output style. One outputs lists of sentences (Hu and Liu, 2004; Blair-Goldensohn et al., 2008; Titov and McDonald, 2008), the other outputs texts consisting of ordered sentences (Carenini et al., 2006; Carenini and Cheung, 2008; Lerman et al., 2009; Lerman and McDonald, 2009). Our work lies in the latter category, and a typical summary is shown in Figure 1. Although visual representations such as bar or rader charts



This restaurant offers customers delicious foods and a relaxing atmosphere. The staff are very friendly but the price is a little high.

Figure 1: A typical summary.

are helpful, such representations necessitate some simplifications of information to presentation. In contrast, text can present complex information that can't readily be visualized, so in this paper we focus on producing textual summaries.

One crucial weakness of existing text-oriented summarizers is the poor readability of their results. Good readability is essential because readability strongly affects text comprehension (Barzilay et al., 2002).

To achieve readable summaries, the extracted sentences must be appropriately ordered (Barzilay et al., 2002; Lapata, 2003; Barzilay and Lee, 2004; Barzilay and Lapata, 2005). Barzilay et al. (2002) proposed an algorithm for ordering sentences according to the dates of the publications from which the sentences were extracted. Lapata (2003) proposed an algorithm that computes the probability of two sentences being adjacent for ordering sentences. Both methods delink sentence extraction from sentence ordering, so a sentence can be extracted that cannot be ordered naturally with the other extracted sentences.

To solve this problem, we propose an algorithm that chooses sentences and orders them simultaneously in such a way that the ordered sentences maximize the scores of informativeness and readability. Our algorithm efficiently searches for the best sequence of sentences by using dynamic programming and beam search. We verify that our method generates summaries that are significantly better than the baseline results in terms of ROUGE score (Lin, 2004) and subjective readability measures. As far as we know, this is the first work to

simultaneously achieve both informativeness and readability in the area of multi-document summarization.

This paper is organized as follows: Section 2 describes our summarization method. Section 3 reports our evaluation experiments. We conclude this paper in Section 4.

2 Optimizing Sentence Sequence

Formally, we define a summary $S^* = \langle s_0, s_1, \dots, s_n, s_{n+1} \rangle$ as a sequence consisting of n sentences where s_0 and s_{n+1} are symbols indicating the beginning and ending of the sequence, respectively. Summary S^* is also defined as follows:

$$S^* = \underset{S \in T}{\operatorname{argmax}} [\operatorname{Info}(S) + \lambda \operatorname{Read}(S)] \quad (1)$$

s.t. $\operatorname{length}(S) \leq K$

where $\operatorname{Info}(S)$ indicates the informativeness score of S , $\operatorname{Read}(S)$ indicates the readability score of S , T indicates possible sequences composed of sentences in the target documents, λ is a weight parameter balancing informativeness against readability, $\operatorname{length}(S)$ is the length of S , and K is the maximum size of the summary.

We introduce the informativeness score and the readability score, then describe how to optimize a sequence.

2.1 Informativeness Score

Since we attempt to summarize reviews, we assume that a good summary must involve as many sentiments as possible. Therefore, we define the informativeness score as follows:

$$\operatorname{Info}(S) = \sum_{e \in E(S)} f(e) \quad (2)$$

where e indicates sentiment $e = \langle a, p \rangle$ as the tuple of *aspect* a and *polarity* $p = \{-1, 0, 1\}$, $E(S)$ is the set of sentiments contained S , and $f(e)$ is the score of sentiment e . Aspect a represents a standpoint for evaluating products and services. With regard to restaurants, aspects include *food*, *atmosphere* and *staff*. Polarity represents whether the sentiment is positive or negative. In this paper, we define $p = -1$ as negative, $p = 0$ as neutral and $p = 1$ as positive sentiment.

Notice that Equation 2 defines the informativeness score of a summary as the sum of the score of the sentiments contained in S . To avoid duplicative sentences, each sentiment is counted only

once for scoring. In addition, the aspects are clustered and similar aspects (e.g. *air*, *ambience*) are treated as the same aspect (e.g. *atmosphere*). In this paper we define $f(e)$ as the frequency of e in the target documents.

Sentiments are extracted using a sentiment lexicon and pattern matched from dependency trees of sentences. The sentiment lexicon¹ consists of pairs of *sentiment expressions* and their polarities, for example, *delicious*, *friendly* and *good* are positive sentiment expressions, *bad* and *expensive* are negative sentiment expressions.

To extract sentiments from given sentences, first, we identify sentiment expressions among words consisting of parsed sentences. For example, in the case of the sentence ‘‘This restaurant offers customers delicious foods and a relaxing atmosphere.’’ in Figure 1, *delicious* and *relaxing* are identified as sentiment expressions. If the sentiment expressions are identified, the expressions and its aspects are extracted as aspect-sentiment expression pairs from dependency tree using some rules. In the case of the example sentence, *foods* and *delicious*, *atmosphere* and *relaxing* are extracted as aspect-sentiment expression pairs. Finally extracted sentiment expressions are converted to polarities, we acquire the set of sentiments from sentences, for example, $\langle \textit{foods}, 1 \rangle$ and $\langle \textit{atmosphere}, 1 \rangle$.

Note that since our method relies on only sentiment lexicon, extractable aspects are unlimited.

2.2 Readability Score

Readability consists of various elements such as conciseness, coherence, and grammar. Since it is difficult to model all of them, we approximate readability as the natural order of sentences.

To order sentences, Barzilay et al. (2002) used the publication dates of documents to catch temporally-ordered events, but this approach is not really suitable for our goal because reviews focus on entities rather than events. Lapata (2003) employed the probability of two sentences being adjacent as determined from a corpus. If the corpus consists of reviews, it is expected that this approach would be effective for sentiment summarization. Therefore, we adopt and improve Lapata’s approach to order sentences. We define the

¹Since we aim to summarize Japanese reviews, we utilize Japanese sentiment lexicon (Asano et al., 2008). However, our method is, except for sentiment extraction, language independent.

readability score as follows:

$$\text{Read}(S) = \sum_{i=0}^n \mathbf{w}^\top \phi(s_i, s_{i+1}) \quad (3)$$

where, given two adjacent sentences s_i and s_{i+1} , $\mathbf{w}^\top \phi(s_i, s_{i+1})$, which measures the connectivity of the two sentences, is the inner product of \mathbf{w} and $\phi(s_i, s_{i+1})$, \mathbf{w} is a parameter vector and $\phi(s_i, s_{i+1})$ is a feature vector of the two sentences. That is, the readability score of sentence sequence S is the sum of the connectivity of all adjacent sentences in the sequence.

As the features, Lapata (2003) proposed the Cartesian product of content words in adjacent sentences. To this, we add named entity tags (e.g. LOC, ORG) and connectives. We observe that the first sentence of a review of a restaurant frequently contains named entities indicating location. We aim to reproduce this characteristic in the ordering.

We also define feature vector $\Phi(S)$ of the entire sequence $S = \langle s_0, s_1, \dots, s_n, s_{n+1} \rangle$ as follows:

$$\Phi(S) = \sum_{i=0}^n \phi(s_i, s_{i+1}) \quad (4)$$

Therefore, the score of sequence S is $\mathbf{w}^\top \Phi(S)$. Given a training set, if a trained parameter \mathbf{w} assigns a score $\mathbf{w}^\top \Phi(S^+)$ to an correct order S^+ that is higher than a score $\mathbf{w}^\top \Phi(S^-)$ to an incorrect order S^- , it is expected that the trained parameter will give higher score to naturally ordered sentences than to unnaturally ordered sentences.

We use Averaged Perceptron (Collins, 2002) to find \mathbf{w} . Averaged Perceptron requires an argmax operation for parameter estimation. Since we attempt to order a set of sentences, the operation is regarded as solving the Traveling Salesman Problem; that is, we locate the path that offers maximum score through all n sentences as s_0 and s_{n+1} are starting and ending points, respectively. Thus the operation is NP-hard and it is difficult to find the global optimal solution. To alleviate this, we find an approximate solution by adopting the dynamic programming technique of the Held and Karp Algorithm (Held and Karp, 1962) and beam search.

We show the search procedure in Figure 2. \mathbf{S} indicates intended sentences and \mathbf{M} is a distance matrix of the readability scores of adjacent sentence pairs. $\mathbf{H}^i(\mathbf{C}, j)$ indicates the score of the hypothesis that has covered the set of i sentences \mathbf{C} and has the sentence j at the end of the path,

<p>Sentences: $\mathbf{S} = \{s_1, \dots, s_n\}$ Distance matrix: $\mathbf{M} = [a_{i,j}]_{i=0 \dots n+1, j=0 \dots n+1}$ 1: $\mathbf{H}^0(\{s_0\}, s_0) = 0$ 2: for $i : 0 \dots n - 1$ 3: for $j : 1 \dots n$ 4: foreach $\mathbf{H}^i(\mathbf{C} \setminus \{j\}, k) \in \mathbf{b}$ 5: $\mathbf{H}^{i+1}(\mathbf{C}, j) = \max_{\mathbf{H}^i(\mathbf{C} \setminus \{j\}, k) \in \mathbf{b}} \mathbf{H}^i(\mathbf{C} \setminus \{j\}, k)$ 6: $+ \mathbf{M}_{k,j}$ 7: $\mathbf{H}^* = \max_{\mathbf{H}_n(\mathbf{C}, k)} \mathbf{H}^n(\mathbf{C}, k) + \mathbf{M}_{k, n+1}$</p>
--

Figure 2: Held and Karp Algorithm.

i.e. the last sentence of the summary being generated. For example, $\mathbf{H}^2(\{s_0, s_2, s_5\}, s_2)$ indicates a hypothesis that covers s_0, s_2, s_5 and the last sentence is s_2 . Initially, $\mathbf{H}^0(\{s_0\}, s_0)$ is assigned the score of 0, and new sentences are then added one by one. In the search procedure, our dynamic programming based algorithm retains just the hypothesis with maximum score among the hypotheses that have the same sentences and the same last sentence. Since this procedure is still computationally hard, only the top \mathbf{b} hypotheses are expanded.

Note that our method learns \mathbf{w} from texts automatically annotated by a POS tagger and a named entity tagger. Thus manual annotation isn't required.

2.3 Optimization

The argmax operation in Equation 1 also involves search, which is NP-hard as described in Section 2.2. Therefore, we adopt the Held and Karp Algorithm and beam search to find approximate solutions. The search algorithm is basically the same as parameter estimation, except for its calculation of the informativeness score and size limitation. Therefore, when a new sentence is added to a hypothesis, both the informativeness and the readability scores are calculated. The size of the hypothesis is also calculated and if the size exceeds the limit, the sentence can't be added. A hypothesis that can't accept any more sentences is removed from the search procedure and preserved in memory. After all hypotheses are removed, the best hypothesis is chosen from among the preserved hypotheses as the solution.

3 Experiments

This section evaluates our method in terms of ROUGE score and readability. We collected 2,940 reviews of 100 restaurants from a website. The

	R-2	R-SU4	R-SU9
Baseline	0.089	0.068	0.062
Method1	0.157	0.096	0.089
Method2	0.172	0.107	0.098
Method3	0.180	0.110	0.101
Human	0.258	0.143	0.131

Table 1: Automatic ROUGE evaluation.

average size of each document set (corresponds to one restaurant) was 5,343 bytes. We attempted to generate 300 byte summaries, so the summarization rate was about 6%. We used CRFs-based Japanese dependency parser (Imamura et al., 2007) and named entity recognizer (Suzuki et al., 2006) for sentiment extraction and constructing feature vectors for readability score, respectively.

3.1 ROUGE

We used ROUGE (Lin, 2004) for evaluating the content of summaries. We chose ROUGE-2, ROUGE-SU4 and ROUGE-SU9. We prepared four reference summaries for each document set.

To evaluate the effects of the informativeness score, the readability score and the optimization, we compared the following five methods.

Baseline: employs MMR (Carbonell and Goldstein, 1998). We designed the score of a sentence as term frequencies of the content words in a document set.

Method1: uses optimization without the informativeness score or readability score. It also used term frequencies to score sentences.

Method2: uses the informativeness score and optimization without the readability score.

Method3: the proposed method. Following Equation 1, the summarizer searches for a sequence with high informativeness and readability score. The parameter vector w was trained on the same 2,940 reviews in 5-fold cross validation fashion. λ was set to 6,000 using a development set.

Human is the reference summaries. To compare our summarizer to human summarization, we calculated ROUGE scores between each reference and the other references, and averaged them.

The results of these experiments are shown in Table 1. ROUGE scores increase in the order of Method1, Method2 and Method3 but no method could match the performance of Human. The methods significantly outperformed Baseline ac-

	Numbers
Baseline	1.76
Method1	4.32
Method2	10.41
Method3	10.18
Human	4.75

Table 2: Unique sentiment numbers.

ording to the Wilcoxon signed-rank test.

We discuss the contribution of readability to ROUGE scores. Comparing Method2 to Method3, ROUGE scores of the latter were higher for all criteria. It is interesting that the readability criterion also improved ROUGE scores.

We also evaluated our method in terms of sentiments. We extracted sentiments from the summaries using the above sentiment extractor, and averaged the unique sentiment numbers. Table 2 shows the results.

The references (Human) have fewer sentiments than the summaries generated by our method. In other words, the references included almost as many other sentences (e.g. reasons for the sentiments) as those expressing sentiments. Carenini et al. (2006) pointed out that readers wanted “detailed information” in summaries, and the reasons are one of such piece of information. Including them in summaries would greatly improve summarizer appeal.

3.2 Readability

Readability was evaluated by human judges. Three different summarizers generated summaries for each document set. Ten judges evaluated the thirty summaries for each. Before the evaluation the judges read evaluation criteria and gave points to summaries using a five-point scale. The judges weren’t informed of which method generated which summary.

We compared three methods; Ordering sentences according to publication dates and positions in which sentences appear after sentence extraction (**Method2**), Ordering sentences using the readability score after sentence extraction (**Method2+**) and searching a document set to discover the sequence with the highest score (**Method3**).

Table 3 shows the results of the experiment. Readability increased in the order of Method2, Method2+ and Method3. According to the

	Readability point
Method2	3.45
Method2+	3.54
Method3	3.74

Table 3: Readability evaluation.

Wilcoxon signed-rank test, there was no significance difference between Method2 and Method2+ but the difference between Method2 and Method3 was significant, $p < 0.10$.

One important factor behind the higher readability of Method3 is that it yields longer sentences on average (6.52). Method2 and Method2+ yielded averages of 7.23 sentences. The difference is significant as indicated by $p < 0.01$. That is, Method2 and Method2+ tended to select short sentences, which made their summaries less readable.

4 Conclusion

This paper proposed a novel algorithm for sentiment summarization that takes account of informativeness and readability, simultaneously. To summarize reviews, the informativeness score is based on sentiments and the readability score is learned from a corpus of reviews. The preferred sequence is determined by using dynamic programming and beam search. Experiments showed that our method generated better summaries than the baseline in terms of ROUGE score and readability.

One future work is to include important information other than sentiments in the summaries. We also plan to model the order of sentences globally. Although the ordering model in this paper is local since it looks at only adjacent sentences, a model that can evaluate global order is important for better summaries.

Acknowledgments

We would like to sincerely thank Tsutomu Hirao for his comments and discussions. We would also like to thank the reviewers for their comments.

References

Hisako Asano, Toru Hirano, Nozomi Kobayashi and Yoshihiro Matsuo. 2008. Subjective Information Indexing Technology Analyzing Word-of-mouth Content on the Web. *NTT Technical Review*, Vol.6, No.9.

Regina Barzilay, Noemie Elhadad and Kathleen McKeown. 2002. Inferring Strategies for Sentence Ordering in Multidocument Summarization. *Journal of Artificial Intelligence Research (JAIR)*, Vol.17, pp. 35–55.

Regina Barzilay and Lillian Lee. 2004. Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 113–120.

Regina Barzilay and Mirella Lapata. 2005. Modeling Local Coherence: An Entity-based Approach. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 141–148.

Sasha Blair-Goldensohn, Kerry Hannan, Ryan McDonald, Tyler Neylon, George A. Reis and Jeff Reynar. 2008. Building a Sentiment Summarizer for Local Service Reviews. WWW Workshop NLP Challenges in the Information Explosion Era (NLPiX).

Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pp. 335–356.

Giuseppe Carenini, Raymond Ng and Adam Pauls. 2006. Multi-Document Summarization of Evaluative Text. In *Proceedings of the 11th European Chapter of the Association for Computational Linguistics (EACL)*, pp. 305–312.

Giuseppe Carenini and Jackie Chi Kit Cheung. 2008. Extractive vs. NLG-based Abstractive Summarization of Evaluative Text: The Effect of Corpus Controversiality. In *Proceedings of the 5th International Natural Language Generation Conference (INLG)*, pp. 33–41.

Michael Collins. 2002. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *Proceedings of the 2002 Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 1–8.

Michael Held and Richard M. Karp. 1962. A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics (SIAM)*, Vol.10, No.1, pp. 196–210.

Minqing Hu and Bing Liu. 2004. Mining and Summarizing Customer Reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 168–177.

- Kenji Imamura, Genichiro Kikui and Norihito Yasuda. 2007. Japanese Dependency Parsing Using Sequential Labeling for Semi-spoken Language. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL) Companion Volume Proceedings of the Demo and Poster Sessions*, pp. 225–228.
- Mirella Lapata. 2003. Probabilistic Text Structuring: Experiments with Sentence Ordering. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 545–552.
- 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)*, pp. 514–522.
- Kevin Lerman and Ryan McDonald. 2009. Contrastive Summarization: An Experiment with Consumer Reviews. In *Proceedings of Human Language Technologies: the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), Companion Volume: Short Papers*, pp. 113–116.
- Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, pp. 74–81.
- Jun Suzuki, Erik McDermott and Hideki Isozaki. 2006. Training Conditional Random Fields with Multivariate Evaluation Measures. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL (COLING-ACL)*, pp. 217–224.
- Ivan Titov and Ryan 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: Human Language Technologies (ACL-HLT)*, pp. 308–316.