Opinion Extraction Using a Learning-Based Anaphora Resolution Technique

Nozomi Kobayashi Ryu Iida Kentaro Inui Yuji Matsumoto
Nara Institute of Science and Technology
Takayama, Ikoma, Nara, 630-0192, Japan
{nozomi-k,ryu-i,inui,matsu}@is.naist.jp

Abstract

This paper addresses the task of extracting opinions from a given document collection. Assuming that an opinion can be represented as a tuple ⟨*Subject*, *Attribute*, *Value*⟩, we propose a computational method to extract such tuples from texts. In this method, the main task is decomposed into (a) the process of extracting *Attribute-Value* pairs from a given text and (b) the process of judging whether an extracted pair expresses an opinion of the author. We apply machine-learning techniques to both subtasks. We also report on the results of our experiments and discuss future directions.

1 Introduction

The explosive spread of communication on the Web has attracted increasing interest in technologies for automatically mining large numbers of message boards and blog pages for opinions and recommendations.

Previous approaches to the task of mining a large-scale document collection for opinions can be classified into two groups: the document classification approach and the information extraction approach. In the document classification approach, researchers have been exploring techniques for classifying documents according to semantic/sentiment orientation such as positive vs. negative (e.g. (Dave et al., 2003; Pang and Lee, 2004; Turney, 2002)). The information extraction approach, on the other hand, focuses on the task of extracting elements which constitute opinions (e.g. (Kanayama and Nasukawa, 2004; Hu and Liu, 2004; Tateishi et al., 2001)).

The aim of this paper is to extract opinions that represent an evaluation of a products together with the evidence. To achieve this, we consider our task from the information extraction viewpoint. We term the above task **opinion extraction** in this paper.

While they can be linguistically realized in many ways, opinions on a product are in fact often expressed in the form of an attribute-value pair. An attribute represents one aspect of a subject and the value is a specific language expression that qualifies or quantifies the aspect. Given this observation, we approach our goal by reducing the task to a general problem of extracting four-tuples ⟨*Product*, *Attribute*, *Value*, *Evaluation*⟩ from a large-scale text collection. Technology for this opinion extraction task would be useful for collecting and summarizing latent opinions from the Web. A straightforward application might be generation of radar charts from collected opinions as suggested by Tateishi et al. (2004).

Consider an example from the automobile domain, *I am very satisfied with the powerful engine (of a car)*. We can extract the four-tuple ⟨*CAR, engine, powerful, satisfied*⟩ from this sentence. Note that the distinction between *Value* and *Evaluation* is not easy. Many expressions used to express a *Value* can also be used to express an *Evaluation*. For this reason, we do not distinguish value and evaluation, and therefore consider the task of extracting triplets ⟨*Product*, *Attribute*, *Value*⟩. Another problem with opinion extraction is that we want to get only subjective opinions. Given this setting, the opinion extraction task can be decomposed into two subtasks: extraction of attribute-value pairs related to a product and determination of its subjectivity.

As we discuss in section 3, an attribute and its value may not appear in a fixed expression and may be separated. In some cases, the attribute may be missing from a sentence. In this respect, finding the attribute of a value is similar to finding the missing antecedent of an anaphoric expression. In this paper, we discuss the similarities and differences between opinion extraction and anaphora resolution. Then, we apply a machine learning-based method used for anaphora reso-

lution to the opinion extraction problem and report on our experiments conducted on a domain-restricted set of Japanese texts excerpted from review pages on the Web.

2 Related work

In this section, we discuss previous approaches to the opinion extraction problem. In the pattern-based approach (Murano and Sato, 2003; Tateishi et al., 2001), pre-defined extraction patterns and a list of evaluative expressions are used. These extraction patterns and the list of evaluation expressions need to be manually created. However, as is the case in information extraction, manual construction of rules may require considerable cost to provide sufficient coverage and accuracy.

Hu and Liu (2004) attempt to extract the attributes of target products on which customers have expressed their opinions using association mining, and to determine whether the opinions are positive or negative. Their aim is quite similar to our aim, however, our work differs from theirs in that they do not identify the value corresponding to an attribute. Their aim is to extract the attributes and their semantic orientations.

Taking the semantic parsing-based approach, Kanayama and Nasukawa (2004) apply the idea of transfer-based machine translation to the extraction of attribute-value pairs. They regard the extraction task as translation from a text to a sentiment unit which consists of a sentiment value, a predicate, and its arguments. Their idea is to replace the translation patterns and bilingual lexicons with sentiment expression patterns and a lexicon that specifies the polarity of expressions. Their method first analyzes the predicate-argument structure of a given input sentence making use of the sentence analysis component of an existing machine translation engine, and then extracts a sentiment unit from it, if any, using the transfer component.

One important problem the semantic parsing approach encounters is that opinion expressions often appear with anaphoric expressions and ellipses, which need to be resolved to accomplish the opinion extraction task. Our investigation of an opinion-tagged Japanese corpus (described below) showed that 30% of the attribute-value pairs we found did not have a direct syntactic dependency relation within the sentence, mostly due to ellipsis. For example[1],
$\langle$*dezain-wa*$\rangle_a$ *hen-daga watashi-wa* ϕ*-ga* $\langle$*suki-da*$\rangle_v$
$\langle$*design*$\rangle_a$ *weird I [it]* $\langle$*like*$\rangle_v$
(*The design is weird, but I like it.*)
This type of case accounted for 46 out of 100 pairs that did not have direct dependency relations. To analyze predicate argument structure robustly, we have to solve this problem. In the next section, we discuss the similarity between the anaphora resolution task and the opinion extraction task and propose to apply to opinion extraction a method used for anaphora resolution.

3 Method for opinion extraction

3.1 Analogy with anaphora resolution

We consider the task of extracting opinion tuples $\langle$*Product, Attribute, Value*$\rangle$ from review sites and message boards on the Web dedicated to providing and exchanging information about retail goods. On these Web pages, products are often specified clearly and so it is frequently a trivial job to extract the information for the *Product* slot. We therefore in this paper focus on the problem of extracting $\langle$*Attribute, Value*$\rangle$ pairs.

In the process of attribute-value pair identification for opinion extraction, we need to deal with the following two cases: (a) both a value and its corresponding attribute appear in the text, and (b) a value appears in the text while its attribute is missing since it is inferable form the value expression and the context. The upper half of Figure 1 illustrates these two cases in the automobile domain. In (b), the writer is talking about the "size" of the car, but the expression "size" is not explicitly mentioned in the text. In addition, (b) includes the case where the writer evaluates the product itself. For example, "I'm very satisfied with my car!": in this case, a value expression "satisfied" evaluates the product as a whole, therefore a corresponding attribute does not exists.

For the case (a), we first identify a value expression (*like* in Figure 1) in a given text and then look for the corresponding attribute in the text. Since we also see the case (b), on the other hand, we additionally need to consider the problem of whether the corresponding attribute of the identified value expression appears in the text or not.

The structure of these problems is analogous to that of anaphora resolution; namely, there are exactly two cases in anaphora resolution that have a clear correspondence with the above two cases as illustrated in Figure 1: in (a) the noun phrase (NP) is *anaphoric*; namely, the NP's antecedent appears in the text, and in (b) the noun phrase is *non-anaphoric*. A non-anaphoric NP is either ex-

[1] $\langle\rangle_a$ denotes the word sequence corresponding to the Attribute. Likewise, we also use $\langle\rangle_v$ for the Value.

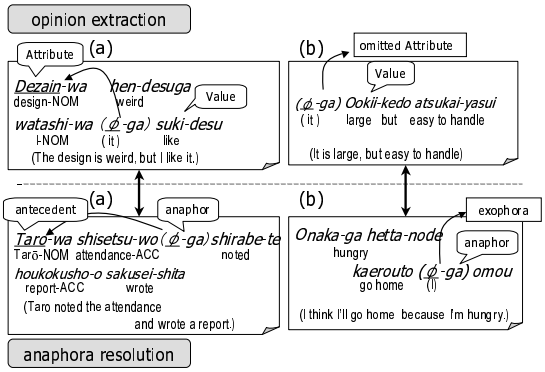

Figure 1: Similarity between opinion extraction and anaphora resolution

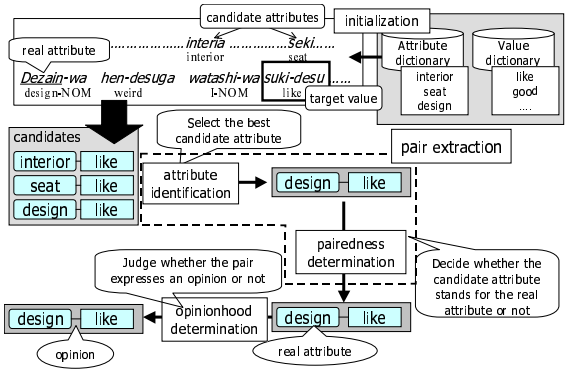

Figure 2: Process of opinion extraction

ophoric (i.e. the NP has an implicit referent) or indefinite. While the figure shows Japanese examples, the similarity between anaphora resolution and opinion extraction is language independent. This analogy naturally leads us to think of applying existing techniques for anaphora resolution to our opinion extraction task since anaphora resolution has been studied for a considerably longer period in a wider range of disciplines as we briefly review below.

3.2 Existing techniques for anaphora resolution

Corpus-based empirical approaches to anaphora resolution have been reasonably successful. This approach, as exemplified by (Soon et al., 2001; Iida et al., 2003; Ng, 2004), is cost effective, while achieving a better performance than the best-performing rule-based systems for the test sets of MUC-6 and MUC-7 [2].

As suggested by Figure 1, anaphora resolution can be decomposed into two subtasks: *anaphoricity determination* and *antecedent identification*. Anaphoricity determination is the task of judging whether a given NP is anaphoric or non-anaphoric. Recent research advances have provided several important findings as follows:

- Learning-based methods for antecedent identification can also benefit from the use of linguistic clues inspired by Centering Theory (Grosz et al., 1995).
- One useful clue for anaphoricity determination is the availability of a plausible candidate for the antecedent. If an appropriate candidate for the antecedent is found in the preceding discourse context, the NP is likely to be anaphoric.

For these reasons, an anaphora resolution model performs best if it carries out the following process in the given order (Iida et al., 2005): (1) **Antecedent identification**: Given an NP, identify the best candidate antecedent for it, and (2) **Anaphoricity determination**: Judge whether the candidate really stands for the true antecedent of the NP.

3.3 An opinion extraction model inspired by analogy with anaphora resolution

As illustrated in Figure 2, an opinion extraction model derived from the aforementioned analogy with anaphora resolution as follows:

1. **Initialization**: Identify attribute and value candidates by dictionary lookup
2. **Attribute identification**: Select a value and identify the best candidate attribute corresponding to the value
3. **Pairedness determination**: Decide whether the candidate attribute stands for the real attribute of the value or not (i.e. the value has no explicit corresponding attribute in the text)
4. **Opinionhood determination**: Judge whether the obtained attribute-value pair[3] expresses an opinion or not

Here, the attribute identification and pairedness determination processes respectively correspond to the antecedent identification and anaphoricity determination processes in anaphora resolution.

Note that our opinion extraction task requires an additional subtask, opinionhood determination — an attribute-value pair appearing in a text does not necessarily constitute an opinion. We elaborate on the notion of opinionhood in section 4.1.

From the above discussion, we can expect that the findings for anaphora resolution mentioned in 3.2 stated above apply to opinion extraction as well. In fact, the information about the candidate

[2]The 7th Message Understanding Conference (1998): www.itl.nist.gov/iaui/894.02/related_projects/muc/

[3]For simplicity, we call a value both with and without an attribute uniformly by the term *attribute-value pair* unless the distinction is important.

attribute is likely to be useful for pairedness determination. We therefore expect that carrying out attribute identification before pairedness determination should outperform the counterpart model which executes the two subtasks in the reversed order. The same analogy also applies to opinionhood determination; namely, we expect that opinion determination is bet performed after attribute determination. Furthermore, our opinion extraction model also can be implemented in a totally machine learning-based fashion.

4 Evaluation

We conducted experiments with Japanese Web documents to empirically evaluate the performance of our opinion extraction model, focusing particularly on the validity of the analogy discussed in the previous section.

4.1 Opinionhood

In these experiments, we define an opinion as follows: *An opinion is a description that expresses the writer's subjective evaluation of a particular subject or a certain aspect of it.*

By this definition, we exclude requests, factual or counter-factual descriptions and hearsay evidence from our target opinions. For example, *The engine is powerful* is an opinion, while a counterfactual sentence such as *If only the engine were more powerful* is not regarded as opinion.

4.2 Opinion-tagged corpus

We created an opinion-tagged Japanese corpus consisting of 288 review articles in the automobile domain (4,442 sentences). While it is not easy to judge whether an expression is a value or an attribute, we asked the annotator to identify attribute and value expressions according to their subjective judgment.

If some attributes are in a hierarchical relation with each other, we asked the annotator to choose the attribute lowest in the hierarchy as the attribute of the value. For example, in *a sound system with poor sound*, only *sound* is annotated as the attribute of the value *poor*.

The corpus contains 2,191 values with an attribute and 420 values without an attribute. Most of the attributes appear in the same sentence as their corresponding values or in the immediately preceding sentence (99% of the total number of pairs). Therefore, we extract attributes and their corresponding values from the same sentence or from the preceding sentence.

4.3 Experimental method

As preprocessing, we analyzed the opinion-tagged corpus using the Japanese morphological analyzer *ChaSen*[4] and the Japanese dependency structure analyzer *CaboCha* [5].

We used Support Vector Machines to train the models for attribute identification, pairedness determination and opinionhood determination. We used the 2nd order polynomial kernel as the kernel function for SVMs. Evaluation was performed by 10-fold cross validation using all the data.

4.3.1 Dictionaries

We use dictionaries for identification of attribute and value candidates. We constructed a attribute dictionary and a value dictionary from review articles about automobiles (230,000 sentences in total) using the semi-automatic method proposed by Kobayashi et al. (2004). The data used in this process was different from the opinion-tagged corpus. Furthermore, we added to the dictionaries expressions which frequently appearing in the opinion-tagged corpus. The final size of the dictionaries was 3,777 attribute expressions and 3,950 value expressions.

4.3.2 Order of model application

To examine the effects of appropriately choosing the order of model application we mentioned in the previous section, we conducted four experiments using different orders (AI indicates attribute identification, PD indicates pairedness determination and OD indicates opinion determination):

Proc.1: OD→PD→AI, Proc.2: OD→AI→PD
Proc.3: AI→OD→PD, Proc.4: AI→PD→OD

Note that Proc.4 is our proposed ordering.

In addition to these models, we adopted a baseline model. In this model, if the candidate value and a candidate attribute are connected via a dependency relation, the candidate value is judged to have an attribute. When none of the candidate attributes have a dependency relation, the candidate value is judged not to have an attribute.

We adopted the tournament model for attribute identification (Iida et al., 2003). This model implements a pairwise comparison (i.e. a match) between two candidates in reference to the given value treating it as a binary classification problem, and conducting a tournament which consists of a series of matches, in which the one that prevails through to the final round is declared the

[4] http://chasen.naist.jp/
[5] http://chasen.org/~taku/software/cabocha/

winner, namely, it is identified as the most likely candidate attribute. Each of the matches is conducted as a binary classification task in which one or other of the candidate wins.

The pairedness determination task and the opinionhood determination task are also binary classification tasks. In Proc.1, since pair identification is conducted before finding the best candidate attribute, we used Soon et al.'s model (Soon et al., 2001) for pairedness determination. This model picks up each possible candidate attribute for a value and determines if it is the attribute for that value. If all the candidates are determined not to be the attribute, the value is judged not to have an attribute. In Proc.4, we can use the information about whether the value has a corresponding attribute or not for opinionhood determination. We therefore create two separate models for when the value does and does not have an attribute.

4.3.3 Features

We extracted the following two types of features from the candidate attribute and the candidate value:

(a) surface spelling and part-of-speech of the target value expression, as well as those of its dependent phrase and those in its depended phrase(s)

(b) relation between the target value and candidate attribute (distance between them, existence of dependency, existence of a co-occurrence relation)

We extracted (b) if the model could use both the attribute and the value information. Existence of a co-occurrence relation is determined by reference to a predefined co-occurrence list that contains attribute-value pair information such as "height of vehicle – low". We created the list from the 230,000 sentences described in section 4.3.1 by applying the attribute and value dictionary and extracting attribute-value pairs if there is a dependency relation between the attribute and the value. The number of pairs we extracted was about 48,000.

4.4 Results

Table 1 shows the results of opinion extraction. We evaluated the results by recall R and precision P defined as follows (For simplicity, we substitute "A-V" for attribute-value pair):

$$R = \frac{\text{correctly extracted A-V opinions}}{\text{total number of A-V opinions}},$$

$$P = \frac{\text{correctly extracted A-V opinions}}{\text{total number of A-V opinions found by the system}}.$$

In order to demonstrate the effectiveness of the information about the candidate attribute, we evaluated the results of pair extraction and opinionhood determination separately. Table 2 shows the results. In the pair extraction, we assume that the value is given, and evaluate how successfully attribute-value pairs are extracted.

4.5 Discussions

As Table 1 shows, our proposed ordering is outperformed on the recall in Proc.3, however, the precision is higher than Proc.3 and get the best F-measure. In what follows, we discuss the results of pair extraction and opinionhood determination.

Pair extraction From Table 2, we can see that carrying out attribute identification before pairedness determination outperforms the reverse ordering by 11% better precision and 3% better recall. This result supports our expectation that knowledge of attribute information assists attribute-value pair extraction. Focusing on the rows labeled "(dependency)" and "(no dependency)" in Table 2, while 80% of the attribute-value pairs in a direct dependency relation are successfully extracted with high precision, the model achieves only 51.7% recall with 61.7% precision for the cases where an attribute and value are not in a direct dependency relation.

According to our error analysis, a major source of errors lies in the attribute identification task. In this experiment, the precision of attribute identification is 78%. A major reason for this problem was that the true attributes did not exist in our dictionary. In addition, a major cause of error in the pair determination stage is cases where an attribute appearing in the preceding sentence causes a false decision. We need to conduct further investigations in order to resolve these problems.

Opinionhood determination Table 2 also shows that carrying out attribute identification followed by opinionhood determination outperforms the reverse ordering, which supports our expectation that knowing the attribute information aids opinionhood determination.

While it produces better results, our proposed method still has room for improvement in both precision and recall. Our current error analysis has not identified particular error patterns — the types of errors are very diverse. However, we need to at least address the issue of modifying the feature set to make the model more sensitive to modality-oriented distinctions such as subjunctive and conditional expressions.

Table 1: The precision and the recall for opinion extraction

procedure		value with attribute		value without attribute		attribute-value pairs	
baseline	precision	60.5%	(1130/1869)	10.6%	(249/2340)	32.8%	(1379/4209)
	recall	51.6%	(1130/2191)	59.3%	(249/420)	52.8%	(1379/2611)
	F-measure	55.7		21.0		40.5	
Proc.1	precision	47.3%	(864/1828)	21.6%	(86/399)	42.7%	(950/2227)
	recall	39.4%	(864/2191)	20.5%	(86/420)	36.4%	(950/2611)
	F-measure	43.0		21.0		39.3	
Proc.2	precision	63.0%	(1074/1706)	38.0%	(198/521)	57.1%	(1272/2227)
	recall	49.0%	(1074/2191)	47.1%	(198/420)	48.7%	(1272/2611)
	F-measure	55.1		42.0		52.6	
Proc.3	precision	74.9%	(1277/1632)	29.1%	(151/519)	63.8%	(1373/2151)
	recall	55.8%	(1222/2191)	36.0%	(151/420)	52.6%	(1373/2611)
	F-measure	64.0		32.2		57.7	
Proc.4	precision	80.5%	(1175/1460)	30.2%	(150/497)	67.7%	(1325/1957)
	recall	53.6%	(1175/2191)	35.7%	(150/420)	50.7%	(1325/2611)
	F-measure	64.4		32.7		58.0	

Table 2: The result of pair extraction and opinionhood determination

	procedure	precision		recall	
pair extraction	baseline (dependency)	71.1%	(1385/1929)	63.2%	(1385/2191)
	PD→AI	65.3%	(1579/2419)	72.1%	(1579/2191)
	AI→PD	76.6%	(1645/2148)	75.1%	(1645/2191)
	(dependency)	87.7%	(1303/1486)	79.6%	(1303/1637)
	(no dependency)	51.7%	(342/ 662)	61.7%	(342/ 554)
opinionhood determination	OD	74.0%	(1554/2101)	60.2%	(1554/2581)
	AI→OD	82.2%	(1709/2078)	66.2%	(1709/2581)

5 Conclusion

In this paper, we have proposed a machine learning-based method for the extraction of opinions on consumer products by reducing the problem to that of extracting attribute-value pairs from texts. We have pointed out the similarity between the tasks of anaphora resolution and opinion extraction, and have applied the machine learning-based method designed for anaphora resolution to opinion extraction. The experimental results reported in this paper show that identifying the corresponding attribute for a given value expression is effective in both pairedness determination and opinionhood determination.

References

K. Dave, S. Lawrence, and D. M. Pennock. 2003. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proc. of the 12th International World Wide Web Conference*, pages 519–528.

B. J. Grosz, A. K. Joshi, and S. Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226.

M. Hu and B. Liu. 2004. Mining and summarizing customer reviews. In *Proc. of the Tenth International Conference on Knowledge Discovery and Data Mining*, pages 168–177.

R. Iida, K. Inui, H. Takamura, and Y. Matsumoto. 2003. Incorporating contextual cues in trainable models for coreference resolution. In *Proc. of the EACL Workshop on the Computational Treatment of Anaphora*, pages 23–30.

R. Iida, K. Inui, Y. Matsumoto, and S. Sekine. 2005. Noun phrase coreference resolution in Japanese base on most likely antecedant candidates. *Journal of Information Processing Society of Japan*, 46(3). (in Japanese).

H. Kanayama and T. Nasukawa. 2004. Deeper sentiment analysis using machine translation technology. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 494–500.

N. Kobayashi, K. Inui, Y. Matsumoto, K. Tateishi, and T. Fukushima. 2004. Collecting evaluative expressions for opinion extraction. In *Proc. of the 1st International Joint Conference on Natural Language Processing*, pages 584–589.

S. Murano and S. Sato. 2003. Automatic extraction of subjective sentences using syntactic patterns. In *Proc. of the Ninth Annual Meeting of the Association for Natural Language Processing*, pages 67–70. (in Japanese).

V. Ng. 2004. Learning noun phrase anaphoricity to improve coreference resolution: Issues in representation and optimization. In *Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 152–159.

B. Pang and L. Lee. 2004. A sentiment education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 271–278.

W. M. Soon, H. T. Ng, and D. C. Y. Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544.

K. Tateishi, Y. Ishiguro, and T. Fukushima. 2001. Opinion information retrieval from the Internet. In *IPSJ SIGNL Note 144-11*, pages 75–82. (in Japanese).

K. Tateishi, T. Fukushima, N. Kobayashi, T. Takahashi, A. Fujita, K. Inui, and Y. Matsumoto. 2004. Web opinion extraction and summarization based on viewpoints of products. In *IPSJ SIGNL Note 163*, pages 1–8. (in Japanese).

P. D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424.