NAIST-IS-DD0461010

# Doctoral Dissertation

# Opinion Mining from Web documents: Extraction and Structurization

Nozomi Kobayashi

March 23, 2007

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Nozomi Kobayashi

Thesis Committee:

| | |
|---|---|
| Professor Yuji Matsumoto | (Supervisor) |
| Professor Shunsuke Uemura | (Member) |
| Professor Kiyohiro Shikano | (Member) |
| Associate Professor Kentaro Inui | (Member) |

# Opinion Mining from Web documents: Extraction and Structurization[*]

Nozomi Kobayashi

## Abstract

This dissertation deals with the task of extracting customer opinions from web documents. This task is the key component of opinion mining, which allows Web users to retrieve and summarize people's opinions scattered over Web documents.

Our aim is to develop a method for extracting opinions, that represent evaluation of consumer products, in a structured form. In this dissertation, we approaches opinion extraction by addressing the following two unexplored issues: how to define the task of opinion extraction and how to extract the structured opinions.

Based on a corpus study, we define an opinion unit consisting of a quadruple, that is, the opinion holder, the subject being evaluated (Subject), the part or the attribute in which it is evaluated (Aspect), and the evaluation that expresses a positive or negative assessment (Evaluation). We use this definition as a basis for our opinion extraction task.

For the second issue, we divide this task into two subtasks: (a) extracting relations between subjects/aspects and evaluations, and (b) extracting relations between subjects/aspects and aspects. Firstly, we consider the approach to extract these relations using a list of expressions which possibly describe subjects, aspects or evaluations. We propose a semi-automatic method for collecting aspect/evaluation expressions, which uses particular co-occurrence patterns of subjects, aspects and evaluations. Our semi-automatic method can collect these

[*] Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0461010, March 23, 2007.

expressions much more efficiently than manual collection. Secondly, we discuss a method for extracting aspect-evaluation relations using dictionaries of aspect and evaluation. We point out that finding the aspect of an evaluation is similar to finding the missing antecedent of an ellipsis, and introduce a machine learning-based method used for anaphora resolution to this task. By using anaphora resolution techniques, we achieve nearly 20 point improvement in F-measure compared with a baseline model. Thirdly, we approach the task for extracting aspect-evaluation relations and aspect-aspect relations without relying on an aspect dictionary. We approach two subtasks using methods which combine contextual clues and context-independent statistical clues. We show that the models using the contextual clues show nearly 10 % improvement in both recall and precision, and the contextual clues learned in a domain are effective in other domains, which indicates the portability of our proposed model.

**Keywords:**

opinion extraction, opinion mining, sentiment analysis, relation extraction

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| ACC | Accusative case (“*o*”) |
| CONJ | Conjunctive |
| DAT | Dative case (“*ni*”) |
| GEN | Genitive case (“*no*”) |
| MOD | Modifier |
| NOM | Nominative case (“*ga*”) |
| TOP | Topic (“*wa*”) |

# Chapter 1

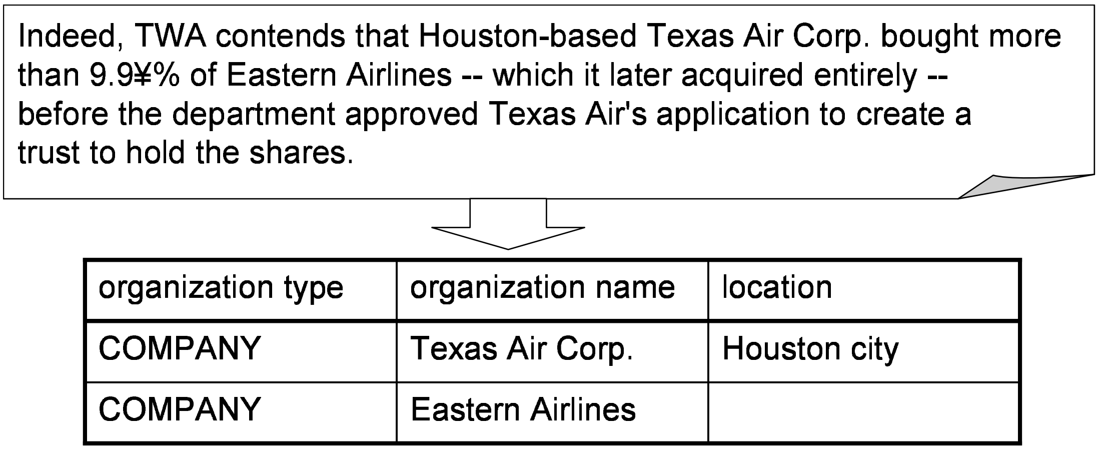
# Introduction

Extracting information from news articles and other texts is an important application task for natural language processing technology. In the 90s, the Message Understanding Conference (MUC) [1], greatly influenced to research in information extraction. Information extraction in the MUC refers to automatic methods for creating a structured representation of information extracted from texts. More specifically, information extraction systems can identify particular types of entities (e.g. organization names, location names) and relationships between entities (e.g. located_at) in texts for storage in a structured database as shown in Figure 1.1. A number of systems have been developed for extracting facts about terrorism, management succession, and so on.

In the past few years, web documents are receiving great attention as a new medium that describes individual experiences and opinions, as symbolized by the new word such as “Blog journalism” or “Consumer generated media (CGM)”. This situation is generating increasing interest in technologies for automatically extracting or analyzing personal opinions from web documents such as posts on message board and weblogs. Such technologies can be an alternative to traditional questionnaire-based social or customer research and would also benefit Web users who seek reviews on certain consumer products of their interest.

Previous approaches to the task of mining a large-scale document collection of customer opinions (or reviews) can be classified into two approaches: text classification and information extraction approaches. In the former, researchers

[1] http://www.itl.nist.gov/iad/894.02/related_projects/muc/index.html



Indeed, TWA contends that Houston-based Texas Air Corp. bought more than 9.9¥% of Eastern Airlines -- which it later acquired entirely -- before the department approved Texas Air's application to create a trust to hold the shares.

| organization type | organization name | location |
|---|---|---|
| COMPANY | Texas Air Corp. | Houston city |
| COMPANY | Eastern Airlines | |

Figure 1.1. An example of factual information extraction

have been exploring techniques for classifying documents or passages according to semantic/sentiment orientation such as positive vs. negative [Dave *et al.*, 2003; Pang and Lee, 2004; Turney, 2002, etc.]. The latter, on the other hand, focuses on the task of extracting opinions consisting of information about particular aspects of interest and the corresponding semantic orientation in a structured form from unstructured text data. In contrast to sentiment classification, opinion extraction in general aims at producing richer information useful for in-depth analysis of opinions, which has recently been taken on by a growing research community [Hu and Liu, 2004; Kanayama and Nasukawa, 2004; Popescu and Etzioni, 2005, etc.].

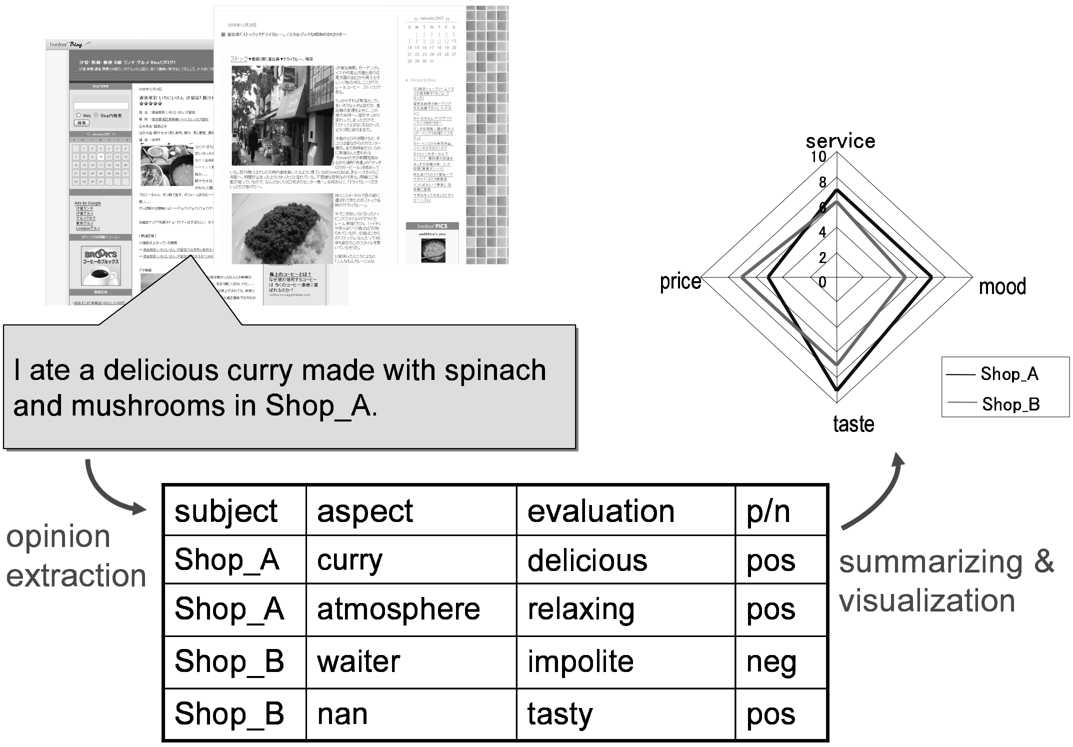
## 1.1 Objective and goal

We approach the task of *opinion mining* as shown in Figure 1.2. We decompose the problem of opinion mining into the following series of subtasks:

1. Extraction of opinions in a structured form

2. Determination of semantic orientation: To each extracted opinion, we assign a semantic orientation: positive, negative, or neutral.

3. Classification of extracted opinions: classify the extracted opinions into predefined categories (for example, "*delicious curry*" and "*tasty nan*" may be



I ate a delicious curry made with spinach and mushrooms in Shop_A.

| subject | aspect | evaluation | p/n |
|---|---|---|---|
| Shop_A | curry | delicious | pos |
| Shop_A | atmosphere | relaxing | pos |
| Shop_B | waiter | impolite | neg |
| Shop_B | nan | tasty | pos |

Figure 1.2. Opinion mining

classified as the same category "positive taste")

4. Visualization of the opinions: This step visualizes the opinions by creating the radar chart [Tateishi *et al.*, 2004] or bar chart [Liu *et al.*, 2005] representing the ratio between the numbers of positive and negative opinions.

This dissertation deals with the first subtask: extracting opinions from web documents. We refer to this extraction task as *opinion extraction*. For the second task, several techniques have already been reported by many researchers [Hatzivassiloglou and McKeown, 1997; Takamura *et al.*, 2005; Wilson *et al.*, 2005, etc.], and we would incorporate these techniques to our overall system. The third task will be solved by using a dictionary which contains the information of relation between each expression and its corresponding category, whereas it is not easy to assume such dictionary to various domains. Developing a domain-independent method for this problem is still an open task. However, the techniques for the first and the second tasks are very useful as an application. The fourth task, visualization of the opinions, is a straightforward application of opinion extraction, therefore it is not a bottle-neck.

We therefore focus on the opinion extraction task, which is one of the key components of opinion mining. For this task, we need to consider the following unexplored issues:

1. How to define the task of opinion extraction

2. How to extract the structured opinions

There are many types of "opinion" such as beliefs, evaluations, requests, etc. Thus, first of all, we need to set up the task of opinion extraction. Previous work does not sufficiently discuss how customer reviews can be best structured. We address the issue and define the opinion extraction tasks in terms of relationship between an aspect of a given evaluation (i.e. aspect) and its value (i.e. evaluation). Given this definition of information extraction, we can structure the opinions in an easy-to-extract manner.

The second issue is to develop a method for extracting these structured opinions. Existing methods for opinion extraction tend to rely on relatively simple proximity-based or pattern-based techniques. However, these pattern-based techniques are not enough to extract opinions because these patterns can apply to the case where all constituents of opinion appear in a sentence. As we will demonstrate, most of the opinion constituents do not have a direct syntactic dependency relation within a sentence, mostly due to ellipsed arguments. For coping with this issue, we introduce a machine learning-based method for extracting opinion constituents.

Firstly, we propose a method for extracting aspect-evaluation relations using dictionaries of aspect and evaluation expressions. We show that our method outperforms the simple pattern-based model both in recall and precision. Secondly, we approach to the method for extracting aspect-evaluation relations without relying on an aspect dictionary. We propose a domain-independent method which combines two different clues: The contextual information and the context-independent information. We apply the same framework to extract the relation between aspects. Our experimental result shows that our proposed model, which does not use an aspect dictionary, achieved a result comparable to the model using the aspect dictionary. The result also shows that the contextual clues learned from a given domain are effective in other domains.

## 1.2 Outline of this dissertation

The rest of this dissertation is organized as follows. In Chapter 2, we describe the opinion units that we aim to extract and the typology of opinion constituents we use. We then set up an opinion extraction task based on our corpus study. In Chapter 3, we outline several previous research efforts on opinion mining, in particular, attempts to develop a method of extracting opinions. In Chapter 4, we propose a semi-automatic method that uses particular co-occurrence patterns of subjects, aspects and evaluations to collect the evaluative expressions. Chapter 5 and Chapter 6 describe our methods for automatically extracting opinions from texts using a machine learning-based approach. In Chapter 5, we deal with aspect–evaluation pair extraction using dictionaries of aspect and evaluation. In Chapter 6, we describe a method for extracting aspect–evaluation relations and aspect–aspect relations without relying on an aspect dictionary. Chapter 7 concludes our work and presents the future directions.

# Chapter 2

# Designing the Task of Opinion Extraction and Structurization

## 2.1 Introduction

In the traditional information extraction task, factual information such as terrorism or management successions has been focused as the target of the extraction. In the factual information extraction task, the target of the extraction is a restricted set of entities. Researchers have paid considerable attention on the problem of named entity (people names, place names, temporal expressions and certain types of numerical expressions) extraction task. In the opinion extraction task, on the other hand, it is unclear what should be extracted, since the opinions include subjective expressions on various topics. Previous work does not sufficiently discuss how customer reviews reported in web documents can be structurized. In this chapter, we reconsider the issue and define an opinion extraction task based on our corpus study.

## 2.2 Constituents of an opinion

Our present goal is to build a computational model to extract opinions from weblog posts in a form like the following:

> *who* evaluates *how* on *which aspects* of *which subjects*

Here we assume that the *subject* of an evaluation is either a consumer product (e.g. a cellular phone model) or a corporate body (e.g. a restaurant, manufacturer, etc.) in a given domain of interest. Given the passage presented in Figure 2.1, for example, one of the opinions we want to extract is the information that *the writer* feels that *the colors* of *pictures* taken with *Powershot* (product) are *great*.

As suggested by this example, we consider it reasonable to start with an assumption that most evaluative opinions expressed in web documents can be structurized as a frame composed of the following constituents:

**Opinion holder** A person who is making an evaluation (usually, either the author or an unspecified person)

**Subject** A named entity (product or company) of a given particular class of interest (e.g. a car model name in the automobile domain).

**Part** A part, member or related object of the subject with respect to which evaluation is made (*engine*, *interior*, etc. in the automobile domain)

**Attribute** An attribute (of a part) of the subject with respect to which evaluation is made (*size, color, design*, etc.)

**Evaluation** An evaluative or subjective phrase used to express an evaluation or the opinion holder's mental/emotional attitude (*good, poor, powerful, stylish, (I) like, (I) am satisfied*, etc.)

**Condition** A condition under which the evaluation applies (*driving on winding roads, when traveling with a family*, etc.)

**Support** An objective fact or experience described as a supporting factor of the evaluation (*weights nearly 1,500 kg*, etc.)

According to this typology, the example text given in Figure 2.1 has eight constituents, *the writer* (opinion holder), *Powershot* (subject), *pictures* (part), *colors* (attribute), *great* (evaluation), *easy to grip* (evaluation), *when flash is used* (condition), and *body has a grip handle* (support), which we consider to constitute two units of opinion as illustrated in the figure. We call each unit an *opinion unit*.

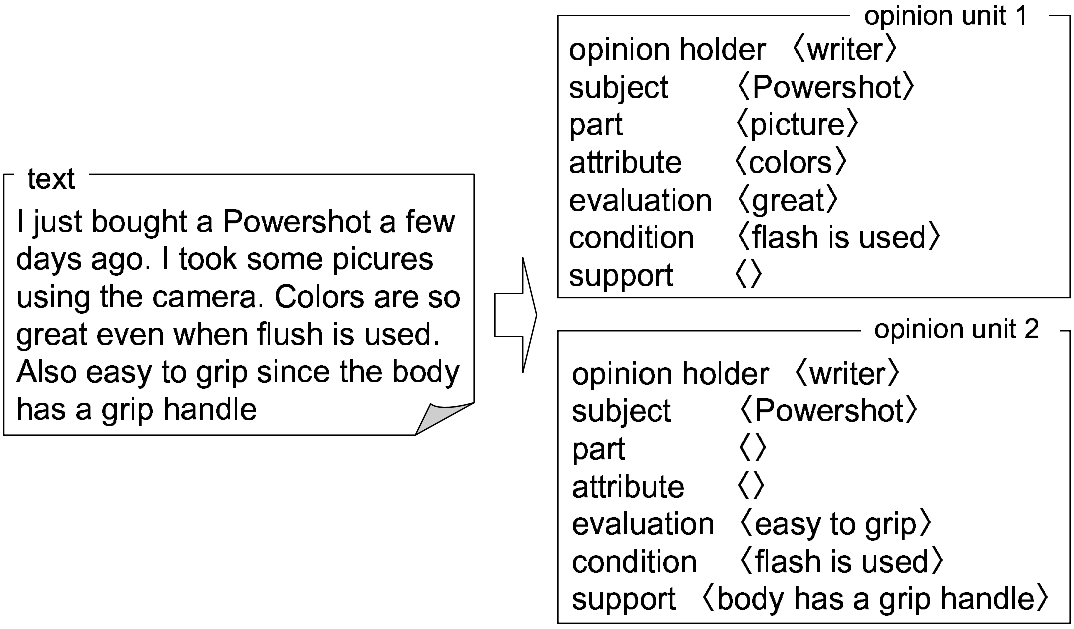

text
I just bought a Powershot a few days ago. I took some picures using the camera. Colors are so great even when flush is used. Also easy to grip since the body has a grip handle
opinion unit 1
opinion holder ⟨writer⟩
subject ⟨Powershot⟩
part ⟨picture⟩
attribute ⟨colors⟩
evaluation ⟨great⟩
condition ⟨flash is used⟩
support ⟨⟩
opinion unit 2
opinion holder ⟨writer⟩
subject ⟨Powershot⟩
part ⟨⟩
attribute ⟨⟩
evaluation ⟨easy to grip⟩
condition ⟨flash is used⟩
support ⟨body has a grip handle⟩

Figure 2.1. Exraction of opinion units

Under this assumption, opinion extraction can be defined as the task of filling the above slots for each of the evaluations expressed in a given text collection. Two issues then immediately arise. First, it is necessary to make sure that the definition of the opinion units is clear enough for human annotators to be able to carry out the task with sufficient accuracy. Second, all the slots might not consist of simple expressions in that the filler of a part or attribute slot may have a hierarchical structure in itself. For example, "*the leather cover of the steering wheel*" refers to a part of a part of a car. In theory, such a hierarchical chain can be of any length, which may affect the feasibility of the task. For these issues, we built a corpus annotated with the above information and investigated the feasibility of the task. In what follows, we report on the results of our corpus study and design an opinion extraction task based on them.

## 2.3 Corpus study

We first collected 116 Japanese weblog posts in the restaurant domain by randomly sampling from the posts classified under the "gourmet" category on the livedoor blog site [1]. A majority of the sampled posts included descriptions about the writer's experience and evaluation regarding certain restaurants.

We asked two annotators to annotate them independently of each other according to the above definition. One annotator (S) was a doctoral program student engaged in research on opinion extraction, while the other was an adult person (A) who did not have expertise in natural language processing.

In the annotation process, every evaluative or subjective phrase was considered as a *candidate* evaluation phrase and, for each candidate evaluation phrase, each annotator was asked to judge whether it constituted an opinion unit or not. If judged yes, a candidate evaluation phrase was associated with a new opinion unit whose slots were to be filled. For each opinion unit, the annotators were asked to identity the opinion holder and the subject while being allowed to leave other slots open if there are nothing appropriate.

Here, we slightly simplified the structure of an opinion unit — we merged the part and attribute slots together. We call the merged slot the *aspect* slot. We did it because we had found, in our preliminary trial, that it is considerably difficult to make a clear distinction between parts and attributes. For example, the phrase *buffet* is used to refer to a physical object belonging to a restaurant, while it may also be used to refer to a function of a restaurant. In the former case, the phrase *buffet* should fill the part slot, while, in the latter, it may be interpreted as an attribute. However, this kind of judgment is sometimes extremely hard.

Consequently, the annotators filled the opinion holder, subject and evaluation slots obligatorily, while filling the aspect, condition and support slots optionally. They were also asked to identify hierarchical relations between aspects (e.g. *noodle* and its *volume*), if any. The following is an example of the annotation in restaurant domain. The underlined expressions denote evaluations, phrases marked with {} are subjects, and $\langle\rangle$ indicate aspects. $n$ in $\langle\rangle_{n-m}$ indicate the correspondence of the evaluation which has the same number, and $m$ expresses the depth of the hierarchy.

> 前から気になっていた抹茶チーズケーキを食べに『{京はやしや}$_{a,b,c,d}$』へ。
> (*We went to "{Kyohayashiya}$_a$" to eat Matcha cheese cake which I*

[1] http://blog.livedoor.com/

*interested in.*)

$\langle$お茶$\rangle_{a-1}$がけっこう$\underline{\text{高い}}_{a}$ので、単品で。

(*We ordered the cake only, since* $\langle$*the tea*$\rangle_{a-1}$ *is* $\underline{\textit{expensive}}_{a}$*.*)

私は$\langle$抹茶チーズケーキ$\rangle_{b-2}$，友人は$\langle$千代の白パフェ$\rangle_{c-1,d-2}$を注文。

(*I ordered a* $\langle$*Matcha cheese cake*$\rangle_{b-2}$*, and my friend ordered a* $\langle$*Chiyo no shiro parfait*$\rangle_{c-2}$*.*)

さて，$\langle$お味$\rangle_{b-1,d-1}$のほうは...

(*Well, how's* $\langle$*the taste*$\rangle_{b-1,d-1}$*...*)

めっちゃ$\underline{\text{おいしい〜}}_{b}$！

(*Very* $\underline{\textit{delicious}}_{b}$*!*)

友人のも$\langle$抹茶$\rangle_{c-1}$が$\underline{\text{濃く}}_{c}$て$\underline{\text{おいし〜}}_{d}$☆

(*And the friend's also* $\underline{\textit{delicious}}_{d}$ *since its contains* $\underline{\textit{rich}}_{c}$ $\langle$*matcha*$\rangle_{c-1}$*.*)

For the above example, we can extract following four opinion units ($\langle$opinion holder, subject, $\langle$aspect$\rangle$, evaluation$\rangle$)

- $\langle$writer, 京はやしや, $\langle$お茶$\rangle$, 高い$\rangle$
  ($\langle$*writer, Kyohayashiya,* $\langle$*the tea*$\rangle$*, expensive*$\rangle$)
- $\langle$writer, 京はやしや, $\langle$抹茶チーズケーキ, お味$\rangle$, おいしい$\rangle$
  ($\langle$*writer, Kyohayashiya,* $\langle$*Matcha cheese cake, the taste*$\rangle$*, delicious*$\rangle$)
- $\langle$writer, 京はやしや, $\langle$千代の白パフェ, 抹茶$\rangle$, 濃い$\rangle$
  ($\langle$*writer, Kyohayashiya,* $\langle$*Chiyo no shiro parfait, matcha*$\rangle$*, rich*$\rangle$)
- $\langle$writer, 京はやしや, $\langle$千代の白パフェ, お味$\rangle$, おいしい$\rangle$
  ($\langle$*writer, Kyohayashiya,* $\langle$*Chiyo no shiro parfait, the taste*$\rangle$*, delicious*$\rangle$)

### 2.3.1 Inter-annotator agreement

We then investigated the degree of the inter-annotator agreement. In the task of identifying evaluations, one annotator (A) identified 450 evaluations, while the other (S) 392 evaluations, 329 cases of which got agreement. Two annotators did not identify the same number of evaluation, so we use the following metric for measuring agreement as [Wiebe *et al.*, 2005] do:

$$agr(A||B) = \frac{\text{\# of tags agreed by A and B}}{\text{\# of tags annotated only by A}}$$

This metric corresponds to the recall if A's annotation is always correct, and to precision, if they are reversed. $agr(A||S)$ was 0.73 and $agr(S||A)$ was 0.83, which indicate that the human can identify evaluation at a certain reasonable level. Next, we investigated the inter-annotator agreement of the aspect- and subject-evaluation relations whose evaluation slot had agreement. Annotator (A) identified 328 relations, and (S) identified 346 relations. 295 cases got agreement, and $agr(S||A)$ was 0.90 and $agr(A||S)$ was 0.86, which show that we obtained high consistency. Finally, for the subject- and aspect-aspect relations, annotator (A) identified 296 relations, while (B) identified 293, 233 cases of which got agreement. $agr(S||A)$ was 0.79 and $agr(A||S)$ was 0.80, which show that the human annotators can carry out the task at a certain level of accuracy.

### 2.3.2 Opinion-tagged corpus

Based on these results, we next collected a larger set of weblog posts for four domains, restaurant, automobile, cellular phone and video game, and asked annotator A to annotate them in the same annotation scheme as above.

We collected Japanese weblog posts from the restaurant domain by randomly sampling from the posts classified under the "gourmet" category on the livedoor blog site, and for the automobile, cellular phone, and video game, we collected weblog pages by issuing subject names as queries to a weblog search engine. The results are summarized in Table 2.1. One observation is that, for all the domains, the length of the hierarchical chains of aspects are longer than two (Subj-Asp-Asp-Eval) in only less than 10% of all the opinion units. From this, we can conclude that hierarchical chains of aspects are unlikely to be too complicated to handle.

The row of "Non-writer Opinion holder" in Table 2.1 shows the number of opinion units whose opinion holder is *not* the writer of the document. The results indicate that when an evaluative description is found, its opinion holder is highly likely to be the writer of the document, which suggests that identification of opinion holder is not a hard problem.

Table 2.1 also shows that the occurrence of supports and conditions is not as frequent as one may expect. While we are aware that supports and conditions, if any, may well provide important information for opinion analysis, we should

Table 2.1. Statistics of opinion-tagged corpora (Rest: restaurant, Auto: automobile, Phone: cellular phone and Game: video game)

| | Rest | Auto | Phone | Game |
|---|---|---|---|---|
| articles | 1,356 | 564 | 481 | 361 |
| sentences | 21,666 | 14,005 | 11,638 | 6,448 |
| Asp-Eval | 3,692 | 943 | 965 | 521 |
| Asp-Asp | 1,426 | 280 | 296 | 221 |
| Subj-Asp | 2,632 | 877 | 850 | 451 |
| Non-writer Opinion holder | 95 | 17 | 22 | 2 |
| Support | 68 | 86 | 80 | 95 |
| Condition | 113 | 86 | 76 | 41 |
| # of opinion units | 4,267 | 1,519 | 1,518 | 775 |
| Subj-Eval | 575 | 576 | 553 | 243 |
| Subj-Asp-Eval | 2,314 | 736 | 768 | 351 |
| Subj-Asp-Asp-Eval | 1,065 | 175 | 172 | 127 |
| other | 313 | 32 | 25 | 54 |

conclude from the statistics that it is practical to put a higher priority of research on the task of filling the other four slots: opinion holder, subject, aspect and evaluation.

### 2.3.3 Task definition

In this dissertation, based on this corpus study, we consider an opinion extraction task as follows:

> Given a text collection, extract opinions and structure them in the form of quadruple ⟨*Opinion holder, Subject, Aspect, Evaluation*⟩, where *Subject* and *Evaluation* are obligatory while *Aspect* is optional and may have a hierarchical chain.

The followings are examples.

(1) *I hear that the ipod is very good.*

→⟨*unspecified person, ipod,* $\phi$*, good*⟩

(2) *I got Canon G3 and am amazed at the quality of photos.*

→⟨*the writer, Canon G3,* ⟨*photos, quality*⟩*, be amazed*⟩

(3) *Nokia 6800 has a nice color screen.*

→⟨*the writer, Nokia 6800, color screen, nice*⟩

### 2.3.4 The task addressed in this dissertation

Our opinion extraction task is now recast as the task of filling the slots of ⟨Opinion holder, Subject, Aspect, Evaluation⟩. Among these slots, we put aside the task of filling the opinion holder slot in this dissertation because the filler of this slot is highly likely to be the writer of the document as noted in Section 2.3.2. Furthermore, we consider identification of candidate subjects (e.g. product names) as a separate task, which has been intensively studied over a decade as the task of named entity recognition. We assume the availability of state-of-the-art models of named entity recognition.

Based on these discussions, in this dissertation we address following tasks:

i) Identifying aspect/evaluation candidate expressions
We propose a semi-automatic method for collecting aspect/evaluation candidate expressions in Chapter 4.

ii) Extracting ⟨Subject, Aspect, Evaluation⟩ triplets by decomposing the problem into two extraction tasks: Aspect-evaluation pair extraction and aspect-aspect (or subject-aspect) relation extraction. We describe these subtasks in Chapter 5 and Chapter 6.

## 2.4 Related work

One of the early work taking the information extraction approach to opinion extraction is reported by Tateishi *et al.* [2001]. The task they consider is extraction of ⟨*Subject, Aspect, Evaluation*⟩ triplets in our terms, and its semantic orientation which is binary-valued, either *positive* or *negative*. However, the reliability of the data used in their experiments is not demonstrated. This researches focus on Japanese text, while in other language, Hu and Liu [2004] consider the task of extracting ⟨*Aspect, Sentence, Semantic-orientation*⟩ triples in our terminology, where *Sentence* is the sentence that includes the *Aspect*, and *Semantic-orientation* is binary-valued, either positive or negative. Our task setting can be considered as a refinement of theirs in that we consider hierarchical chains of aspects, which may be filled with phrases even from separate sentences, and we also consider the evaluation slot to be filled with an evaluation phrase. They annotated semantic orientations and aspects to review articles, however, the reliability of the data is not demonstrated.

Perhaps, our task setting is closest to the one considered by Popescu and Etzioni [2005] and Yi *et al.* [2003]. Popescu and Etzioni consider the task of extracting ⟨Aspect, Evaluation, Semantic-orientation⟩, and reported the inter-annotator agreement of the triplets in their evaluation data. However, their papers lack discussion of task formulation based on corpus studies.

To our best knowledge, one of the most extensive corpus studies in this field is being conducted by the MPQA project [Wiebe *et al.*, 2005] [2]. In this corpus, individual expressions are marked that correspond to explicit mentions of private states[3] (e.g. "*The U.S. fear a spill-over*"), speech events (e.g. "*Sue said that she would be home late*"), and expressive subjective elements (e.g. "*to put it mildly*", "*what an idiot*"). However, their concerns are not necessarily focused on the types of customer opinions we consider, and they annotate newspaper articles, which presumably exhibit a quite different distributions from web documents.

## 2.5 Summary

In this chapter, we discussed the task of structuring opinions, and introduced the opinion units consisting of four constituents: ⟨Opinion holder, Subject, Aspect, Evaluation⟩. We then set up an opinion extraction task based on our corpus study. The rest of this dissertation, we consider the task of ⟨Subject, Aspect, Evaluation⟩. Before going into the main topics, we outline several previous research efforts on opinion mining in the next chapter.

[2] This corpus is available at http://nrrc.mitre.org/NRRC/02_results/mpqa.html

[3] "private state" is a general term that covers opinions, beliefs, thoughts, feelings, emotions, goals, evaluations, and judgments [Wiebe, 2002]

# Chapter 3

# Previous work on opinion mining

In this chapter, we outline previous research efforts in opinion mining. First, we describe the document classification approach which determine semantic orientation of the documents (or their sentences). And then, we explain various approaches to opinion extraction in Section 3.2.

## 3.1 Classifying opinions into positive/negative

Semantic orientation determination is a task of determining whether a sentence or document has either positive or negative orientation. There are two early works attempting this task reported by [Pang *et al.*, 2002] and [Turney, 2002]. The approaches for this task can be decomposed into two approaches: the unsupervised approach [Turney, 2002] and the supervised approach [Pang *et al.*, 2002].

### 3.1.1 Unsupervised approach to sentiment classification

Turney [Turney, 2002; Turney and Littman, 2002] predicts the semantic orientation of the documents based on the average semantic orientation of the adjective phrases and adverb phrases appearing in the documents. In his model, sentiment orientation $SO$ of the phrase $ph$ is estimated as follows.

$$SO(ph) = PMI(ph, pos\_words) - PMI(ph, neg\_words)$$

where $pos\_words$ represents pre-defined positive words such as "*excellent, good*", and $neg\_words$ represents pre-defined negative words such as "*poor, bad*". The pointwise mutual information (PMI) between *ph* and *words* is defined as follows [Church and Hanks, 1989]:

$$PMI(ph, words) \quad = \quad log_2 \frac{p(ph\&words)}{p(ph)p(words)}$$

where $p(ph, words)$ is the probability that *ph* and *words* co-occur. If the words are statistically independent, the probability that they co-occur is given by the product $p(ph)p(words)$. The ratio between $p(ph, words)$ and $p(ph, words)$ is a measure of the degree of statistical dependence between *phrase* and *words*. He calculated PMIs based on the number of web pages returned by search engines, when the pair of the phrase and the word is queried.

Given the semantic orientation of each phrase, the document is classified as positive if the average of the $SO(phrases) > 0$ and negative otherwise.

### 3.1.2 Supervised approach to sentiment classification

Another approach to sentiment classification is based on the supervised machine learning-based method. The task of sentiment classification can be considered as a text categorization (i.e. text classification) task in which texts are classified into one of several predefined categories using information from training texts. In the text categorization task various machine learning methods have been applied, and they have proven successful [Sebastiani, 2002]. The same methods have been applied to the sentiment classification task by many researchers [Pang *et al.*, 2002; Yu and Hatzivassiloglou, 2003; Mullen and Collier, 2004; Matsumoto *et al.*, 2005, etc.]. In the supervised approach, the learning process is driven by the knowledge of the categories (positive/negative, in this task) and of the training instances that belong to them. In this task, online review articles are often used as training and evaluation data, because in review articles, reviewers often summarize their overall sentiment with a rating indicator, such as a number of stars. Therefore, we do not need manual-annotation of the document for supervised learning or evaluation purposes.

Pang *et al.* [2002] examined with three machine learning methods: naive bayes classification, maximum entropy classification and support vector machines. As the features, they used unigrams, bigrams, and so on, which are used in the traditional categorization task. They pointed out that it is difficult to classify a document's sentiment orientation by using only word information, since there are many words indicative of opposite sentiments in the target documents. Based on these findings, they introduced the task of sentence subjectivity classification, to discard the objective sentences from the target document [Pang and Lee, 2004]. This approach is based on an assumption that the semantic orientation of the document relies only on the sentences expressing the writer's subjectivity.

Mullen and Collier [2004] also applied the machine learning method which incorporated several kinds of information as features such as semantic orientation of the evaluative expressions. Their experimental result shows that the model which uses the above information outperforms models which do not use it.

## 3.2 Extracting opinion comprising elements

Other than document level sentiment classification of product reviews, researchers have also been exploring methods for in-depth analysis of opinions by extracting opinions into specific formats.

### 3.2.1 Pattern-based or proximity-based approach

Approaches to the opinion extraction task mainly use simple proximity-or pattern-based techniques. Murano and Sato [2003] and Tateishi *et al.* [2001; 2004] proposed a method which uses pre-defined extraction patterns and a list of evaluative expressions. These extraction patterns and the list of evaluation expressions need to be created manually. For example, they used syntactic patterns such as "⟨*Aspect/Subject*⟩ ga/wa ⟨*Evaluation*⟩" or "⟨*Evaluation*⟩ na ⟨*Aspect*⟩". The former pattern can match the example "⟨*dezain*⟩ *ga* ⟨*yoi*⟩ (*The design is good*)", and the latter can match the example such as "⟨*suteki*⟩ *na* ⟨*dezain*⟩ (*excellent design*)".

Yi *et al.* [2003]'s task is extracting ⟨aspect, evaluation, semantic-orientation⟩ triplets. They identify evaluation expressions using a dictionary which they build using external resources such as WordNet [Fellbaum, 1998] and general inquirer [Stone *et al.*, 1966][1]. WordNet is a lexical database connecting English words/expressions to categories representing their meanings. And general inquirer is a dictionary that contains information about English word senses, including tags that label them as positive, negative, negations, overstatements or understatements. The size of dictionary is approximately 3000 (2500 adjectives and less than 500 nouns). For aspect expressions, they automatically extract these expressions using rules and scoring based on the likelihood. To identify relations between aspects and evaluations, they used manually-created patterns. The patterns have following two types:

- (target, verb, source)
  For example, (*the camera*, *like*, "") and (*the digital zoom*, *be*, *too grainy*) are matched the pattern.

- (adjective, target)
  (*good quality*, *photo*) is matched, for example.

Hu and Liu [2004] defined opinion extraction as the task of extracting opinion sentences which contain one or more aspects and one or more evaluation expressions[2]. Instead of using a dictionary, Hu and Liu [2004] approached the task by filtering out non-aspect candidates using aspect expressions automatically extracted from another large documents.

Popescu and Etzioni [2005] consider the task of extracting ⟨aspect, opinion-phrase, semantic-orientation⟩, where opinion phrases is an adjective, noun, verb or adverb phrase representing customer opinions. They also start at identifying aspects automatically acquired using Web-based information extraction system KnowItAll [Etzioni *et al.*, 2004]. KnowItAll utilizes a set of eight domain-independent extraction patterns (e.g. "NP1 *such as* NP2") to generate candidate facts. Next, the system assigns a probability to each candidate using a Naive Bayes classifier. The Naive Bayes classifier uses the pointwise mutual information used in Turney [2002] as a binary feature.

Then, Popescu and Etzioni identify opinions as follows:

[1] http://www.wjh.harvard.edu/~inquirer/

[2] In [Hu and Liu, 2004], they used adjectives as evaluation expressions

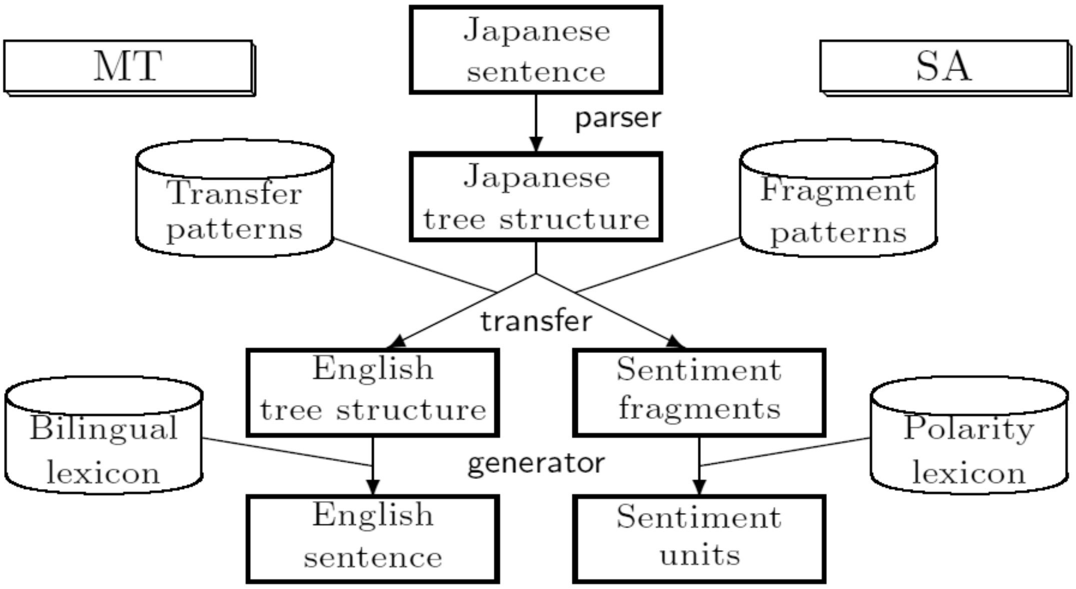


Figure 3.1. The concept of Kanayama *et al.*'s approach (Kanayama *et al.*, 2004)

1. If an explicit aspect is found in a sentence, their system applies the extraction rules

2. The phrase whose head word has a positive or negative orientation is retained as an opinion phrase

To extract opinion phrases, they use syntactic dependencies based on their intuition that "an opinion phrase associated with a product feature will occur in its vicinity". A similar idea is used in [Kim and Hovy, 2004]. Kim and Hovy used various window sizes (e.g. full sentence, words between opinion holder and the subject) instead of syntactic dependencies.

### 3.2.2 Semantic parsing-based approach

Kanayama and Nasukawa [2004] applied the idea of transfer-based machine translation to the extraction of evaluations and evaluated aspects. They regard the extraction task as translation from a text to a sentiment unit which consists of a sentiment evaluation, a predicate and its arguments. Their idea is to replace translation patterns and bilingual lexicons with sentiment expression patterns and a lexicon that specifies the semantic orientation of each expression. 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 present, using the transfer component.

# Chapter 4

# Collecting evaluative expressions

## 4.1 Introduction

There are various method for extracting opinions in the form of triplet ⟨*Subject, Aspect, Evaluation*⟩ as mentioned in Chapter 2. We consider the approach to extract the opinion units using a list of expressions which possibly describe either evaluated subjects, focused aspects or evaluations (referred to subject expressions, aspect expressions, and evaluation expressions, hereafter).

If such a list of these sorts of expressions is available, we may be able to realize opinion extraction in two steps:

1. Detecting expressions included in the list
2. Organizing detected expressions into the form: ⟨*Subject, Aspect, Evaluation*⟩.

As mentioned in previous chapter, we consider identification of candidate subjects (e.g. product names) as a separate task, which has been intensively studied over a decade as the task of named entity recognition. Aspect expressions, on the other hand, are common nouns, and tend to be domain-dependent. For example, "*gas mileage*" is an aspect expression in the automobile domain, but is not in the computer domain. Therefore, we should add the expression "*gas mileage*" to the aspect dictionary in the automobile domain, but should not add it to the dictionary in the computer domain. Evaluation expressions are more likely to be used commonly across different domains. However, there are many

Table 4.1. Evaluation dictionaries (Tateishi et al., 2001)

| domain | evaluation expressions |
|---|---|
| domain-independent | *suki* (*like*), *ii* (*good*), *yoi* (*good*), *saikou* (*great*), ... |
| books | *omoshiroi* (*funny*), *meisaku* (*masterpiece*) ... |
| computers | *hayai* (*fast*), *omoi* (*slow*), *fuantei* (*unstable*), ... |

expressions used for expressing writer's 'evaluation', so it can be expensive to manually create an exhaustive list of expressions. This indicates that how to build an exhaustive dictionary for each domain inexpensively is an important issue.

For this issue, we explored how to accelerate the process of collecting aspect and evaluation expressions by applying a text mining technique. In this chapter, we propose a semi-automatic method that uses particular co-occurrence patterns of subjects, aspects and evaluations, used in the information extraction field. We then empirically evaluate the effectiveness of the semi-automatic method comparing with manual collection.

## 4.2 Related work

Tateishi et al. [2001] proposed a method for extracting opinions using extraction patterns and manually-created dictionaries which include evaluative expressions. They created a dictionary for each domain as shown in Table 4.1.

Murano and Sato [2003] also used a manually created dictionary, however, it is quite expensive to manually create an exhaustive list of expressions for many domains, because dictionaries tend to be domain-dependent. To solve this problem, we propose a method for reducing the cost of creating a list of evaluative expressions: Aspect expressions and evaluation expressions.

There have also been several techniques developed for acquiring subjective words. Riloff and Jones [1999] proposed a method for learning both a list of extraction patterns and a domain-specific semantic lexicon simultaneously. This algorithm needs a set of unlabeled text and pre-defined seed words for the semantic lexicon of interest. In [Riloff *et al.*, 2003], they apply the above method and an improved method proposed in [Thelen and Riloff, 2002], to obtain subjective nouns.

We also try to collect evaluative expressions using a bootstrapping algorithm. Riloff *et al.* collect the extraction patterns and semantic lexicon simultaneously, while we fix the extraction patterns and collect aspect and evaluation expressions. The advantage of automatic collection of extraction patterns is that this method can acquire various extraction patterns, however it has the drawback that some patterns may wrongly generate many expressions we do not want. Checking new acquired expressions is easier than checking the patterns newly extracted, we therefore propose a method using a set of patterns created previously.

## 4.3 Collecting expressions using co-occurrence patterns

Opinions can be linguistically realized in many ways. One of the typical forms would be:

*⟨Aspect⟩ of ⟨Subject⟩ is ⟨Evaluation⟩.*

We use such typical patterns (referred to co-occurrence patterns, hereafter) to collect evaluative expressions. For example, applying the above coocurrence pattern to

(1) *the leather seat of Product_X is comfortable*

we can learn that "*the leather seat*" may be an aspect expression and "*comfortable*" an evaluative expression. However,

(2) *My apartment is near the station.*

also match the above pattern wrongly. To avoid such errors, we introduce the acquired subject/aspeect/evaluation expressions as the constraints. That is, if we have already known that "*comfortable*" is an evaluative expression, we can reason that "*leather seat*" is more likely to be an aspect expression. Based on this idea, we impose a constraint that at least one slot should be filled by the known subject/aspect/evaluation expressions.

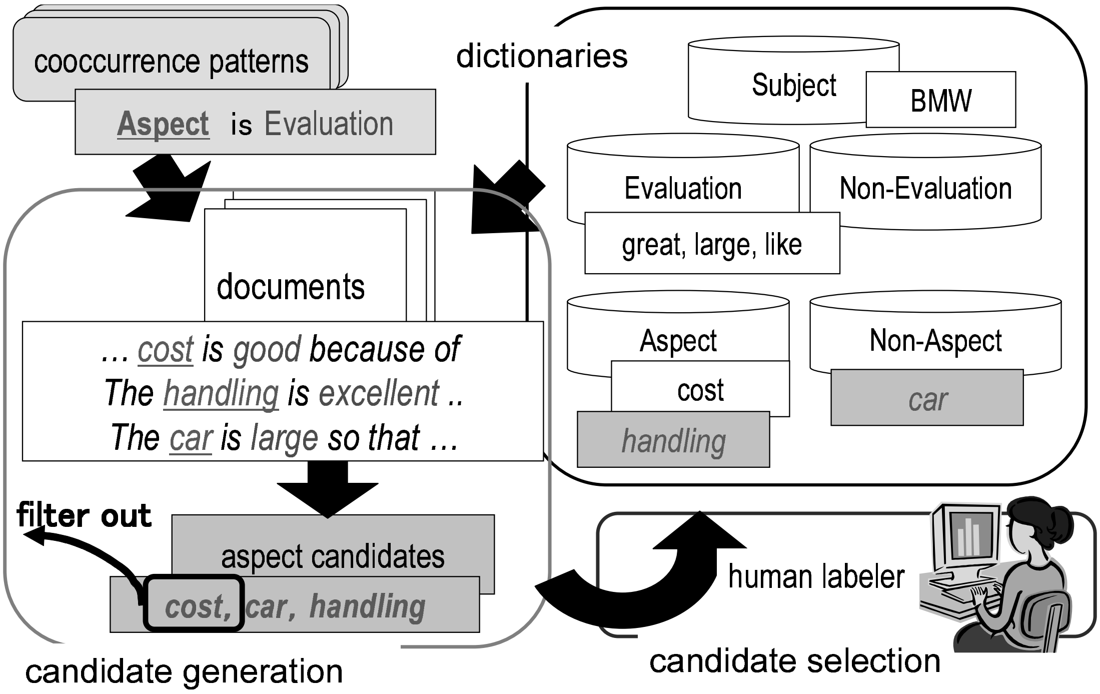


Figure 4.1. Semi-automatic process of collecting aspect/evaluation expressions

Figure 4.1 illustrates the process of collecting aspect/evaluative expressions. The overall process consists of repeated cycles of *candidate generation* followed by *candidate selection*. In each cycle, the candidate generation step automatically produces a ranked list of candidates for either aspect or evaluation expressions using coocurrence patterns and the current dictionaries of subject, aspect and evaluation expressions.

In the candidate selection step, a human judge selects correct aspect/evaluation expressions from the list and add them to the dictionaries. Updates of the dictionaries may allow the candidate generation step to produce different candidates. Repeating this cycle makes both the aspect and evaluation dictionaries richer in each cycle.

### 4.3.1 Candidate generation

We explain the process of candidate generation along Figure 4.1. We describe co-occurrence patterns as follows:

$\langle \underline{Aspect} \rangle$ is $\langle Evaluation \rangle$

In this notation, we assume that $\langle Evaluation \rangle$ corresponds to an already known evaluation expression and the underlined slot $\langle \underline{Aspect} \rangle$ denotes an expression that can be taken as a candidate of an aspect expression. Note that we need to predefine the co-occurrence patterns.

If our document collection includes sentences like (3), we can obtain "*handling*" and "*cost*" as candidates for aspect expressions. $\langle \rangle_a$ denotes the word sequence corresponding to the aspect slot of the co-occurrence pattern. Likewise, we also use $\langle \rangle_e$ for the evaluation slot [1].

(3) … $\langle \underline{cost} \rangle_a$ *is* $\langle good \rangle_e$ *because of* …
… $\langle \underline{the\ handling} \rangle_a$ *is* $\langle excellent \rangle_e$ …

Here we must note that such co-occurrence patterns may also generate non-evaluative candidates as in the following case, from which a candidate expression "*car*" is extracted:

(4) … *The* $\langle \underline{car} \rangle_a$ *is* $\langle large \rangle_e$ *so that…*

To reduce the noise in the extraction, we introduce the filtering method as follows:

1. Filtering using part-of-speech information
   To reduce noise in the extraction, we specify the applicability condition of each pattern based on part-of-speech.

2. Filtering using statistics-based score
   we also use a statistics-based scoring function to rank extracted candidates and provide the human judge with only a limited number of highly ranked candidates.

[1] Likewise, $\langle \rangle_s$ in Figure 4.2 indicates the subject slot

3. Filtering already registered expressions
   To reduce the labor of manual checking of such non-evaluative expressions, we first filter out candidates that have already been registered either in the aspect and evaluation dictionaries. For this purpose, each dictionary is designed to keep expressions that have been judged as evaluative expressions in an earlier cycle as well as non-evaluative expressions. In case of Figure 4.1, "*cost*" is filtered out because it is already registered as an aspect expression.

The details of the scoring function we used in the experiments will be given in Section 4.4.1.

### 4.3.2 Candidate selection

In the candidate selection step, a human judge labels an arbitrary number of highly ranked candidates and register them into the dictionaries. In Figure 4.1, given two candidates "*handling*" and "*car*", the human labeler has judged the former as an aspect expression and the latter as a non-aspect expression. As the result, "*handling*" is added to the aspect dictionary and "*car*" is added to the non-aspect dictionary.

## 4.4 Experiments

We conducted experiments with Japanese Web documents in two domains, automobile and video game (simply game, hereafter), to empirically evaluate the effectiveness of our method compared to a manual collection method. These domains have their own characteristics. In the automobile domain, there are many aspect shared among different car models, in the video game domain, on the other hand, there are many aspect expressions only appeared in certain genre (Role-playing, Fighting, etc.) or particular series.

We used the following time periods for the collection step: 7.5 hours for the automobile domain and 5 hours for the video game domain. These time periods mean the time spent for the manual checking. In the manual collection, the time means that the time human annotated aspect and evaluation expressions using an annotation tool. In the experiments, we hired a person as the examiner who had no knowledge about the technical details of our method.

### 4.4.1 Semi-automatic collection of aspect/evaluation expressions

In this section, we explain the experimental settings for semi-automatic collection.

#### Input data

We collected 15,000 reviews (230,000 sentences) from several review sites on the Web for the car domain and 9,700 reviews (90,000 sentences) for the game domain. We analyzed these documents using the Japanese morphological analyzer ChaSen[2] and the Japanese dependency structure analyzer CaboCha[3].

#### Initial dictionaries

We show the numbers of the expressions used for initial dictionaries.

- Subject dictionary
  In the input data we used, subjects are explicitly written at the top of the article. So we collect these subjects expressions, and made the subject dictionary.
  We collected 389 expressions for automobile domain (e.g. *"BMW", "TOYOTA"*) and 660 expressions for the game domain (e.g. *"Dark Chronicle", "Seaman"*).

- Aspect dictionary
  For the seed set of aspect expressions, we manually chose the following 7 expressions for both domains that considered to be usable across different domains:

  > *nedan* (cost), *kakaku* (price), *sâbisu* (service), *seinou* (performance), *kinou* (function), *sapôto* (support), *dezain* (design).

[2] http://chasen.naist.jp/hiki/
[3] http://chasen.org/˜taku/software/cabocha/

- Evaluation dictionary
  For the seed set of evaluation expressions, we used an existing thesaurus and dictionaries to manually collect those that were considered domain-independent, obtaining 247 expressions, most of which were adjectives. The following are examples of them:

  > *yoi* (good), *kirei* (beautiful), *akarui* (bright), *kiniiru* (like / favorite), *takai* (high), *chiisai* (small)

**Co-occurrence patterns**

We preliminarily tested various co-occurrence patterns against another set of documents collected from the domain of mobile computers. Note that we tested the co-occurrence patterns to another documents in the mobile computer domain. This enables us to investigate if the patterns tuned in the domain work well in the other domain. We then selected eight patterns as shown in Figure 4.2 because they appeared relatively frequently and exhibited reasonable precision[4].

The underlined slot⟨_⟩ denotes an expression that can be taken as a candidate expression, if the other slots are filled. For example, pattern 1 「⟨*Evaluation*⟩ ⟨*Subject*⟩」 means that ⟨Evaluation⟩ is extracted as the candidate evaluation candidate, if ⟨Subject⟩ slot is filled by a known subject.

We assume that the scope of extracted expression is content word (or unknown word) in base-phrase (bunsetsu) boundary. However, some expressions span beyond bunsetsu boundaries (e.g. X-ga aru, X-ga nai), so we allow to span beyond the bunsetsu for such cases.

In addition to above patterns, we used another heuristic rule which indicates aspect and evaluation expressions by suffixes. For example, we regard expressions include a suffix "*-sei* (*-ity*) (e.g. *antei-sei* (*stability*))" as a candidate of aspect, if the expressions do not match the patterns.

**Some filtering method**

As mentioned in Section 4.3.1, we introduce following filtering methods to reduce the noise in the extraction:

[4] We use the variable *Product_X* to cite actual example, this slot should be filled by concrete product name

|  |  |  |
|---|---|---|
| Pat.1 | $\langle\underline{\textit{Evaluation}}\rangle$-MOD | $\langle\textit{Subject}\rangle$ |
| e.g. | $\langle\underline{\textit{shibutoi}}\rangle_e$ | $\langle\textit{Product\_1}\rangle_s$ |
|  | stubborn | Product_1 |
|  | (...stubborn Product_1...) |  |

|  |  |  |
|---|---|---|
| Pat.2 | $\langle\underline{\textit{Evaluation}}\rangle$-MOD | $\langle\textit{Aspect}\rangle$ |
| e.g. | $\langle\underline{\textit{yasuppoi}}\rangle_e$ | $\langle\textit{dezain}\rangle_a$ |
|  | cheap | design |
|  | (...cheap design...) |  |

|  |  |  |
|---|---|---|
| Pat.3 | $\langle\textit{Evaluation}\rangle$-MOD | $\langle\underline{\textit{Aspect}}\rangle$ |
| e.g. | $\langle\textit{subarashii}\rangle_e$ | $\langle\underline{\textit{handoringu}}\rangle_a$ |
|  | great | handling |
|  | (...great handling...) |  |

|  |  |  |
|---|---|---|
| Pat.4 | $\langle\textit{Subject}\rangle$-no | $\langle\underline{\textit{Aspect}}\rangle$ |
| e.g. | $\langle\textit{Product\_3}\rangle_s\textit{-no}$ | $\langle\underline{\textit{dezain}}\rangle_a$ |
|  | Product_3-of | design |
|  | (the design of Product_3) |  |

|  |  |  |
|---|---|---|
| Pat.5 | $\langle\textit{Aspect}\rangle$-{ga,etc.} | $\langle\underline{\textit{Evaluation}}\rangle$ |
| e.g. | $\langle\textit{nennpi}\rangle_a\textit{-ga}$ | $\langle\underline{\textit{yoi}}\rangle_e$ |
|  | gas mileage-NOM | great |
|  | (the gas mileage is great) |  |

|  |  |  |
|---|---|---|
| Pat.6 | $\langle\underline{\textit{Aspect}}\rangle$-{ga,etc.} | $\langle\textit{Evaluation}\rangle$ |
| e.g. | $\langle\underline{\textit{interia}}\rangle_a\textit{-ga}$ | $\langle\textit{yoi}\rangle_e$ |
|  | interior-NOM | nice |
|  | (the interior is nice) |  |

|  |  |  |  |
|---|---|---|---|
| Pat. 7 | $\langle\textit{Subject}\rangle$-no | $\langle\underline{\textit{Aspect}}\rangle$- {wa,etc.} | $\langle\textit{Evaluation}\rangle$ |
| e.g. | $\langle\textit{Product\_1}\rangle_s$-no | $\langle\underline{\textit{interia}}\rangle_a$-wa | $\langle\textit{kirei.}\rangle_e$ |
|  | Product_1-of | interior-TOP | beautiful. |
|  | (the interior of Product_1 is beautiful.) |  |  |

|  |  |  |  |
|---|---|---|---|
| Pat. 8 | $\langle\textit{Subject}\rangle$-no | $\langle\textit{Aspect}\rangle$-{wa,etc.} | $\langle\underline{\textit{Evaluation}}\rangle$ |
| e.g. | $\langle\textit{Product\_2}\rangle_s$-no | $\langle\textit{enjine}\rangle_a$-wa | $\langle\underline{\textit{sizuka}}\rangle_e$ |
|  | Product_2-of | engine-TOP | quiet |
|  | (the engine of Product_2 is quiet.) |  |  |

Figure 4.2. The used co-occurrence patterns

### Filtering using part-of-speech

To specify the part-of-speech we need to extract, we analyzed 50 expressions for each aspect and evaluation expressions extracted from the message boads in computer domain.

Most of the aspect expressions are nouns (42 expressions), and the remains are unknown words (e.g. HDD). Based on this observation, we extract only unknown words, single nouns exept for numerical expressions, and compound nouns as aspect candidates.

The candidates of evaluation expressions are the following:

> nominal adjectivals 18, adjectives 13, nouns 9, verbs 6, sahen-verbs 3, unknown words 1

Common nouns are frequently appeared as evaluation expressions, however, there is a risk of false extraction if we take all nouns as the evaluation candidates.

Most of the extracted nouns are appeared the form of "Noun-da" (e.g. *saiaku-da* (worst), *mimizawari-da* (annoying)), so we target nouns if they appear the form of "Noun-da".

As the result, we extract only adjectives, verbs (including *sahen*-verbs), nominal adjectivals and nouns which appear in the form of "Noun-da" for evaluation candidates.

### Scoring

To the extracted expressions using Pat.1 to Pat.3, Pat.5 and Pat.6 in Figure 4.2, we introduce a scoring function based on frequency. With the frequency-based scoring, Pat.4 to Pat.6 are still relatively underconstrained and tend to generate many non-evaluative expressions. We thus introduce another scoring function based on the reliability of the co-occurrence patterns.

1. Scoring based on the term frequency
   Based on the consideration that candidates with a high frequency in the target document collection have the preference, we score the expressions with the frequency of extracted expressions.

Table 4.2. Contingency table

| | y | $\neg y$ | total |
|---|---|---|---|
| $x$ | $freq(x, y) = a$ | $freq(x, \neg y) = b$ | $freq(x)$ |
| $\neg x$ | $freq(\neg x, y) = c$ | $freq(\neg x, \neg y) = d$ | $freq(\neg x)$ |
| total | $freq(y)$ | $freq(\neg y)$ | |

2. Reliability of the co-occurrence patterns

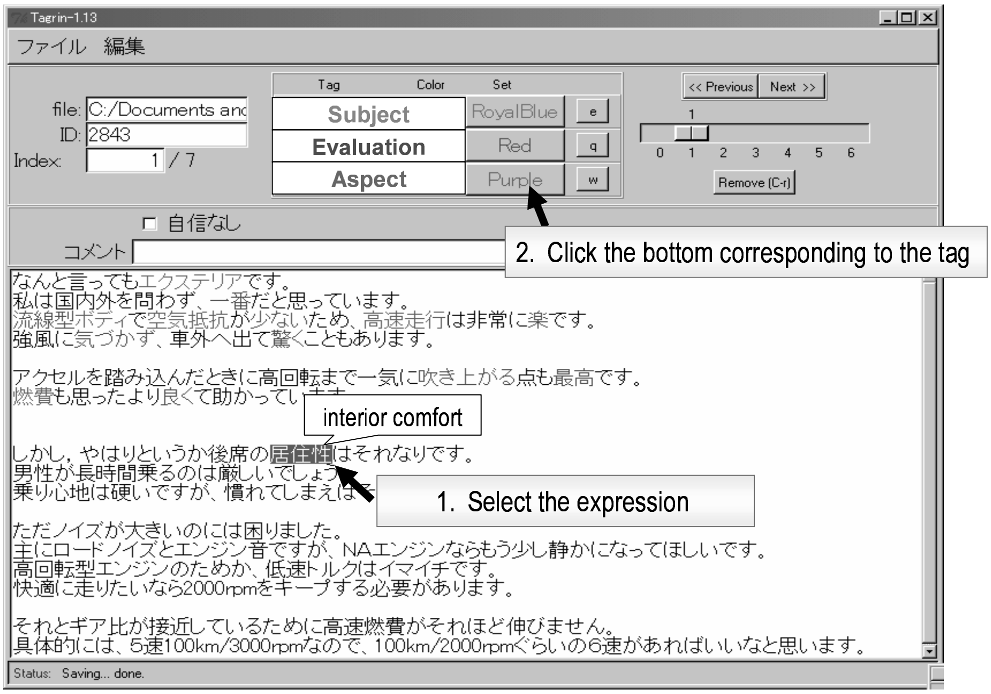
   The other scoring factor is the reliability of clues used for extraction. Suppose that we want to estimate the reliability of an instantiated cooccurrence pattern "$\underline{\langle Aspect\rangle}$ *is low*". If this pattern produces not only correct candidates such as "*cost*" and "*seat position*" but also many non-evaluation candidates such as "*body height*", we can learn from those results that the pattern is not so reliable, presumably less reliable than, say, "$\langle Aspect\rangle$ *is comfortable*" which produces very few non-evaluation candidates. Based on this consideration, we estimate the reliability of an instantiated pattern by a co-occurrence measure. We use log-likelihood ratio[Dunning, 1993] between candidates and evaluation expressions. Given the contingency table descibed in Table 4.2 ($freq(x, y)$ is the number of times $x$ occurred in $y$, and $freq(x)$ is the number of $x$ occurred), we can calculate the log-likelihood ration between a candidate and the evaluation expressions as follows:

$$\begin{aligned} a\ log\ a + b\ log\ b + c\ log\ c + d\ log\ d - \\ (a+b)\ log\ (a+b) - (a+c)\ log\ (a+c) - \\ (b+d)\ log\ (b+d) - (c+d)\ log\ (c+d) + N\ log\ (N) \end{aligned}$$

   where $N = a + b + c + d$.

### 4.4.2 Manual collection of aspect/evaluation expressions

We hired a person as an examiner who had no knowledge about the technical details of our method. Moreover, the examiner had no special knowledge for automobiles and the games. We asked the human examiner to tag aspect and evaluation expressions using an annotation tool shown in Figure 4.3. The annotation process is as follows:



file: C:/Documents anc
ID: 2843
Index: 1 / 7

2. Click the bottom corresponding to the tag

なんと言ってもエクステリアです。
私は国内外を問わず、一番だと思っています。
流線型ボディで空気抵抗が少ないため、高速走行は非常に楽です。
強風に気づかず、車外へ出て驚くこともあります。

アクセルを踏み込んだときに高回転まで一気に吹き上がる点も最高です。
燃費も思ったより良くて助かってい

interior comfort

しかし，やはりというか後席の居住性はそれなりです。
男性が長時間乗るのは厳しいでしょう
乗り心地は硬いですが、慣れてしまえば

1. Select the expression

ただノイズが大きいのには困りました。
主にロードノイズとエンジン音ですが、NAエンジンならもう少し静かになってほしいです。
高回転型エンジンのためか、低速トルクはイマイチです。
快適に走りたいなら2000rpmをキープする必要があります。

それとギア比が接近しているために高速燃費がそれほど伸びません。
具体的には、5速100km/3000rpmなので、100km/2000rpmぐらいの6速があればいいなと思います。

Status: Saving... done.

Figure 4.3. The interface of the tool for manual collection

1. Select the expression considered as evaluation (or aspect)

2. Click the button corresponding to the tag

The human examinar can easily annotate the tags using this tool. After the work, we extracted the annotated expressions automatically, and created the aspect and evaluation dictionaries.

The examiner tagged expressions in 105 reviews (about 5,000 sentences) from the automobile domain and 280 reviews (about 2,000 sentences) from the video game domain. The working time is 7.5 hour for each domain, and the examinar could annotate 105 articles (nearly 5,000 sentences) for automobile domain, and 280 articles (nearly 2,000 sentences) for videogame domain. Those reviews were taken from the same document collections that we used with our semi-automatic method.

It is important to note that while the same person was responsible for both manual collection of evaluative expressions and judgment of our semi-automatic method, we avoided possible conflicts of interest by evaluating our method before manually collecting expressions.

## 4.5 Results and Discussion

### 4.5.1 Collection efficiency

Figures 4 and 5 show the plots of the numbers of collected expressions versus the required time. For the semi-automatic collection, we plot the cumulative number of expressions in each cycle of the collections process. For the manual collection, we plot the cumulative number of expressions collected from each 5 articles.

The figures show that the semi-automatic method is significantly more efficient than the manual collection in collecting the same number of expressions. For example, the semi-automatic method takes only 0.6 hours to collect the first 500 aspect expressions while the manual extraction requires more than 5 hours. We also find that both domains exhibit quite similar tendencies. This indicates that our method is likely to work well in a wide range of domains. Recall that, preliminary to the experiments, we used documents in the mobile computer domain, which was considerably different from the automobile and game domains, to tune the co-occurrence patterns. This suggest that the same set of patterns will work well in other domains.

One problem observed from the results is that the number of extracted expressions does not exhibit convergence. We consider that this tendency is due to the fact that the current semi-automatic method operates greedy. For example, "*engine*" and "*response*" are collected as an aspect expressions, but "*engine response*" which consists of above two aspect expressions, is also collected as an aspect expression. To cope with this problem, we might re-consider the range of the expressions which we need to extract.

### 4.5.2 Coverage

It is also important to see to how successfully the semi-automatically corrected expressions cover the expressions which appear in an unseen data set. We next

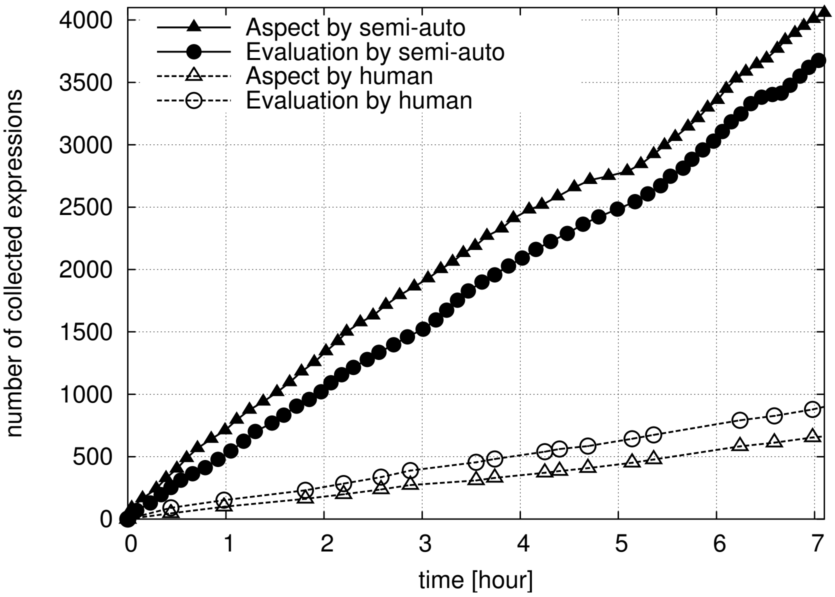


Figure 4.4. Number of collected expressions (automobile)

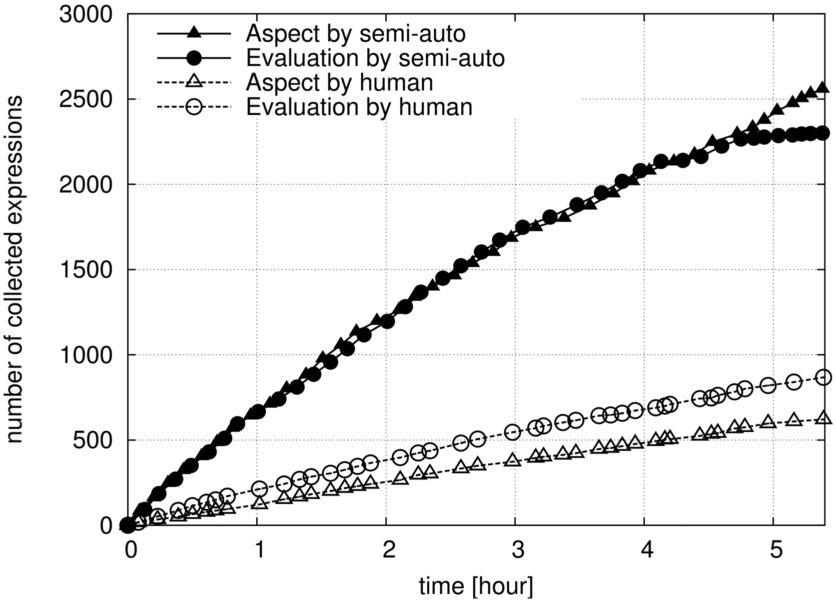


Figure 4.5. Number of collected expressions (game)

Table 4.3. Coverage of collected expressions

| | | aspect | evaluation |
|---|---|---|---|
| automobile | manual | 39.4% (124/315) | 43.3% (164/380) |
| | semi-auto | 65.1% (205/315) | 64.4% (244/380) |
| video game | manual | 43.3% ( 58/134) | 41.4% ( 94/227) |
| | semi-auto | 61.9% ( 83/134) | 51.1% (116/227) |

collected another 100 reviews for each of two domains, and extracted aspect and evaluation expressions manually. Table 4.3 shows the coverage of the semi-automatically and manually collected expressions. In the table, the denominators are the numbers of the aspect or evaluation expressions collected from unseen data, and the numerators are the numbers of expressions covered the semi-automatically and manually collected expressions described in Section 4.4. Table 4.4 shows some examples, where "common" indicates expressions collected commonly in both ways, and "semi-auto" and "manual" are expressions collected only by each method.

For the unseen data, the expressions collected manually cover from 40 % to 50 %, while the semi-automatic collected expressions cover 60%. We find that the semi-automatic collection cover more expressions than manual collection, however, 40 % of the expressions still remain. Table 4.5 shows the main causes that our method has failed to cover the remaining 40%. Note that the denominators are the numbers of the expressions except for the cases where the expressions do not appear in the data we used in the semi-automatic method.

### Filtering using part-of-speech information

As mentioned in Section 4.4.1, we introduced the part-of-speech-based restrictions. Some expressions are filtered out by this restrictions. For example, "shikkari (*tightly*)" is an adverb and "demeritto (*demerit*)" is a noun, but these expressions are considered as evaluation expressions.

Moreover, though we assumed that the evaluation may not be used as aspects and vice versa, actually some expressions are used as aspects. One example is "*hirosa* (*width*)". In a sentence such as

Table 4.4. Examples of collected expressions

| | | both | semi-auto only | manual only |
|---|---|---|---|---|
| automobile | aspect | *sasupenshon*<br>サスペンション<br>(*suspension*)<br>*norigokochi*<br>乗り心地<br>(*ride quality*) | *sîtopozishon*<br>シートポジジョン<br>(*seat position*)<br>*shininsei*<br>視認性<br>(*visibility*) | *inpane*<br>インパネ<br>(*instrument panel*)<br>*torimawashi*<br>取り回し<br>(*treatment*) |
| | evaluation | *hinjaku*<br>貧弱<br>(*poor quality*)<br>*kakkoii*<br>カッコイイ<br>(*cool*) | *shûitsu*<br>秀逸<br>(*brilliant*)<br>*kattarui*<br>かったるい<br>(*tiring*) | *moteamashigimi*<br>持て余し気味<br>(*boring*)<br>*moushibunnnai*<br>申し分ない<br>(*all right*) |
| game | aspect | *sousasei*<br>操作性<br>(*operationality*)<br>*gurafikku*<br>グラフィック<br>(*graphics*) | *sutôrîtenkai*<br>ストーリー展開<br>(*storyline*)<br>*gashitsu*<br>画質<br>(*image quality*) | *pureijikan*<br>プレイ時間<br>(*play time*)<br>*kakushiyouso*<br>隠し要素<br>(*hidden feature*) |
| | evaluation | *subarashii*<br>すばらしい<br>(*excellent*)<br>*miryokuteki*<br>魅力的<br>(*attractive*) | *tyûtohannpa*<br>中途半端<br>(*half-baked*)<br>*tasai*<br>多彩<br>(*various*) | *arienai*<br>ありえない<br>(*impossible*)<br>*oku ga fukai*<br>奥が深い<br>(*deep*) |

⟨*sîto*⟩-*no* *hirosa-ni* ⟨*kangeki*⟩
⟨*seat*⟩-OF *width*-DAT ⟨*be impressed*⟩
(*I was impressed by how wide the seats were.*)

“*hirosa*” is considered as the aspect of the evaluation “*kangaki*”, however, in the case of “*sîto-no hirosa* (*The width of the seats*)”, we can interpret this phrase as “sîto-ga hiroi (*The seats are wide*)”, therefore, “*hirosa*” is considered as an evaluation. We handle this problem in another process which determines whether a candidate aspect (or evaluation) expression is true aspect (or evaluation) or not.

Table 4.5. Categorization of the collecting errors

| | automobile | | game | |
|---|---|---|---|---|
| | aspect | evaluation | aspect | evaluation |
| filtering using POS | 35% (8/23) | 30% (35/115) | 25% (2/8) | 19% (16/85) |
| beyond bunsetsu's | – | 22% (25/115) | 25% (2/8) | 19% (16/85) |

#### Expressions spanning beyond base-phrase

Although the semi-automatic method does not generate candidate expressions spanning beyond base-phrase (bunsetsu) boundaries, some expressions appear beyond bunsetsu boundaries. For example, "*hara-ga tatsu* (*get angry*)" and "*sentô-no shikata* (*how to fight*)" are not collected with this method.

### 4.5.3 Utility of co-occurrence patterns

To evaluate how correctly and exhaustively candidate expressions are collected using fixed extraction patterns, we investigated the performance of eight co-occurrence patterns.

Tables 4.6 and 4.7 shows the usefulness of the patterns, where "number" indicates the number of expressions extracted by the patterns, and "correct/incorrect" indicates the number of evaluation/non-evaluation and aspect/non-aspect expressions. We evaluated the usefulness by precision defined as follows:

$$\text{Precision} = \frac{\text{number of candidates decided as aspect (or evaluation)}}{\text{number of candidates extracted by the pattern}}$$

From Tables 4.6 and 4.7, we can see that both domains exhibit quite similar precision. As mentioned above, the patterns used in the experiments are tuned in different domain (mobile computer domain). Therefore, we can say that the same set of patterns work well in different domains.

Overall, the patterns that extract evaluation expressions outperform the patterns that extract aspects. One reason is that evaluation expressions also cooccur with named entities (e.g. product names, company names, and so on) or general expressions such as "*mono* (thing)".

Table 4.6. Performance of co-occurrence patterns (evaluation extraction pattern)

| | pattern | precision | number | correct/incorrect |
|---|---|---|---|---|
| automobile | Pat.1 | 73.3% | 15 | 11/ 4 |
| | Pat.2 | 81.4% | 1347 | 1097/ 250 |
| | Pat.5 | 69.1% | 4917 | 3398/1519 |
| | Pat.8 | 66.5% | 239 | 159/ 80 |
| video game | Pat.1 | – | – | – / – |
| | Pat.2 | 78.7% | 901 | 709/ 192 |
| | Pat.5 | 82.1% | 2581 | 2119/ 462 |
| | Pat.8 | 93.3% | 15 | 14/ 1 |

Table 4.7. Performance of co-occurrence patterns (aspect extraction pattern)

| | pattern | precision | number | correct/incorrect |
|---|---|---|---|---|
| automobile | Pat.3 | 50.2% | 1136 | 570/ 566 |
| | Pat.4 | 45.6% | 726 | 331/ 395 |
| | Pat.6 | 75.9% | 5225 | 3965/1260 |
| | Pat.7 | 58.2% | 273 | 159/ 114 |
| video game | Pat.3 | 31.5% | 1093 | 344/ 749 |
| | Pat.4 | 62.5% | 40 | 25/ 15 |
| | Pat.6 | 66.2% | 3975 | 2631/1344 |
| | Pat.7 | 56.5% | 23 | 13/ 10 |

## 4.6 Summary

In this chapter, we proposed a semi-automatic method for extracting evaluative expressions based on particular co-occurrence patterns of evaluated subject, focused aspect and evaluation. We reported the experimental results, showing that our semi-automatic method was able to collect aspect and evaluation expressions much more efficiently than manual collection and that the co-occurrence patterns we used in the experiments worked well across different domains.

In the next chapter, we discuss how to extract aspect–evaluation pairs using the dictionaries.

# Chapter 5

# Extracting aspect-evaluation pairs using aspect dictionary

## 5.1 Introduction

In previous chapter, we discussed that if the dictionaries of subject/aspect/evlauation are available, we may be able to realize opinion extraction in two steps:

1. Detecting expressions included in the dictionaries

2. Organizing detected expressions into the form: ⟨*Subject, Aspect, Evaluation*⟩.

In this chapter, we propose a method for extracting aspect–evaluation pairs using domain-specific dictionaries. In particular, we focus on the review articles which can be considered as "clean" data, because the most of that described in the review articles are relevant to the given domain. On these Web pages, products are often specified clearly and it is in many cases a trivial job to extract the information for *Subject* slot. As we will show later, if the product name is given, it is not difficult to detect the *Subject* of the *Evaluation*. We therefore focus on the problem of extracting ⟨*Aspect, Evaluation*⟩ pairs.

## 5.2 Extracting aspect–evaluation pairs

In the process of aspect-evaluation pair identification for opinion extraction, we need to address the following issues. First, arguments of a predicate may not appear in a fixed expression and may be separated. Our analysis of an opinion-tagged Japanese corpus (described in Section 5.3.1) showed that 30% of the aspect-evaluation pairs we found did not have a direct syntactic dependency relation within a sentence, mostly due to ellipses. In the following example, the aspect “*design*” and the evaluation “*like*” are not connected via a dependency relation, since the pronoun (corresponding to “*it*”) is omitted.

$\langle$*dezain-wa*$\rangle_a$ *hen-dato iwarete-iruga watashi-wa* $\phi$ $\langle$*suki*$\rangle_e$
$\langle$*design*-TOP$\rangle_a$ *be-weird said but* *I*-TOP *[it]* $\langle$*like*$\rangle_e$
(*It is said that the design is weird, but I like it.*)

This phenomenon is known as *zero-anaphora*, which is a kind of *anaphora* that refers to the phenomenon that an expression points back to another expression in the preceding context. *Zero-anaphora* is a gap, in a phrase or clause, that has an anaphoric function. The process of identifying this types of anaphoric relation is called *anaphora resolution*.

This leads us to a possibility 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.

Second, as pointed out by Hu and Liu [2004] and Popescu and Etzioni [2005], aspects may not always be explicitly expressed. Let us see two examples from the reviews of the automobile:

“$\langle$*The seat*$\rangle_a$ *is very* $\langle$*comfortable*$\rangle_e$”
“*A* $\langle$*big*$\rangle_e$ *car*”

In the first example, both an evaluation and its corresponding aspect appear in the text, while in the second example, an evaluation appears in the text but its aspect is missing since it is inferable form the evaluation phrase and the context (in this example, “*a big car*” implies the “the size” of the car is “*big*”). For this issue, we introduce a model for determining whether an evaluation has an explicit aspect or not.

Third, recall that evaluation phrases do not always constitute opinions; the target of an evaluation may be neither a subject nor an aspect of a subject of the given domain, and furthermore we want to exclude evaluation phrases appearing,

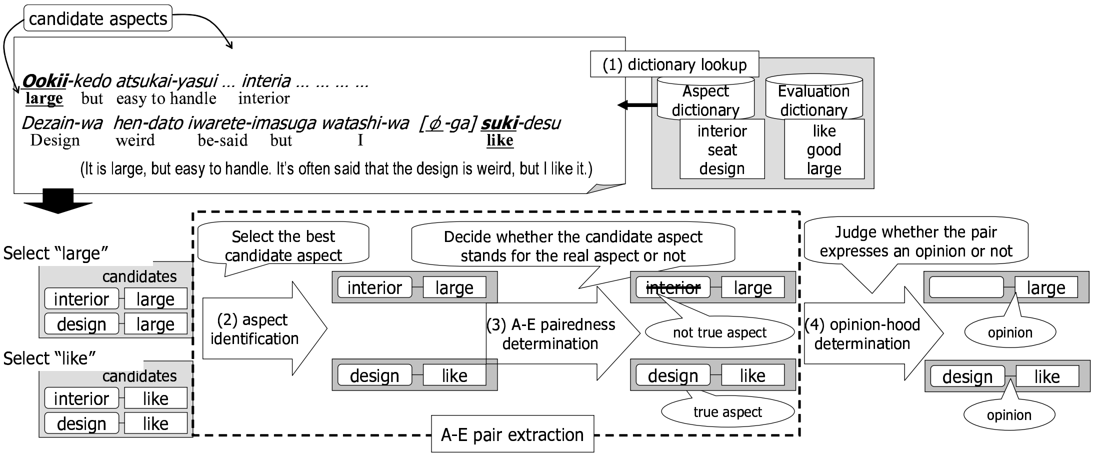

candidate aspects
Ookii-kedo atsukai-yasui ... interia ... ... ... ...
large but easy to handle interior
Dezain-wa hen-dato iwarete-imasuga watashi-wa [ϕ-ga] suki-desu
Design weird be-said but I like
(It is large, but easy to handle. It's often said that the design is weird, but I like it.)
(1) dictionary lookup
Aspect dictionary
interior
seat
design
Evaluation dictionary
like
good
large
Select "large"
candidates
interior large
design large
Select "like"
candidates
interior like
design like
Select the best candidate aspect
(2) aspect identification
interior large
design like
Decide whether the candidate aspect stands for the real aspect or not
(3) A-E pairedness determination
~~interior~~ large
not true aspect
design like
true aspect
A-E pair extraction
Judge whether the pair expresses an opinion or not
(4) opinion-hood determination
large
opinion
design like
opinion

Figure 5.1. Process of opinion extraction

for example, in interrogative and subjunctive sentences. We therefore need to incorporate into our opinion extraction model a classifier for judging whether a given evaluation phrase constitutes an opinion. In the judgment, we expect that the information about the candidate aspect is likely to be useful for the determination. For example,

[1] *kosuto-ga* *takai*
*cost*-NOM *high*
(*the cost is high.*)

[2] *shiyou hindo-ga* *takai*
*frequency of use*-NOM *high*
(*(its) frequency of use is high.*)

These descriptions share the same evaluation expression "*high*". However, [1] is our target opinion, while [2] is not a target opinion because this description describes rather a fact not a writers' subjective evaluation. As this example shows, the plausibility of an evaluation expression to be an opinion changes according to its aspect. From this observation, we expect that carrying out aspect identification before pairedness determination should outperform the counterpart model which executes the two subtasks in the reversed order.

### 5.2.1 Method for opinion extraction

As illustrated in Figure 5.1, we propose an opinion extraction model derived from the aforementioned discussion as follows:

1. **Dictionary lookup**: Assuming that we have domain-specific dictionaries of evaluation and aspect phrases, identify candidate aspects and evaluations by dictionary lookup. In Figure 5.1, "*large*" and "*like*" are evaluation candidates, and "*interior* and "*design*" are aspect candidates.

2. **Aspect identification**: For each candidate evaluation phrase, identify the best candidate aspect. In Figure 5.1, the model identifies the best candidate "*interior*" for the evaluation candidate "*large*". Note that "*large*" may not an explicit aspect.

3. **Aspect-evaluation pairedness determination**: Decide whether the candidate aspect is the true aspect of the evaluation (i.e. the evaluation has an explicit aspect in the text). In this step, we detect whether the evaluation has explicit aspect or not. Note that we do not identify what the omitted aspect is in the case where no explicit aspect is identified. In this example, "*design*" is the true aspect of the evaluation "*like*" and "*interior*" is not the true aspect of the evaluation "*large*".

4. **Opinion-hood determination**: Judge whether the obtained aspect-evaluation pair[1] constitutes an opinion or not. In this example, both "*large*" and "*like*" constitutes an opinion, thus the model judges these are opinions.

We adopt the tournament model [Iida *et al.*, 2003] for aspect identification as shown in Figure 5.2. This model implements a pairwise comparison (i.e., a match) between two candidates in reference to the given evaluation treating it as a binary classification problem, and conducts a tournament which consists of a series of matches, in which the one that prevails through to the final round is declared the winner, namely, it is identified as the most likely candidate aspect. In this figure, CA3 is identified as the most likely candidate aspect. Each of the matches is conducted as a binary classification task in which one or the other candidate wins.

The pairedness determination task and the opinion-hood determination task are also binary classification tasks (whether the evaluation has explicit aspect or

[1] For simplicity, we call an evaluation both with and without an aspect uniformly by the term *aspect-evaluation pair* unless the distinction is important.

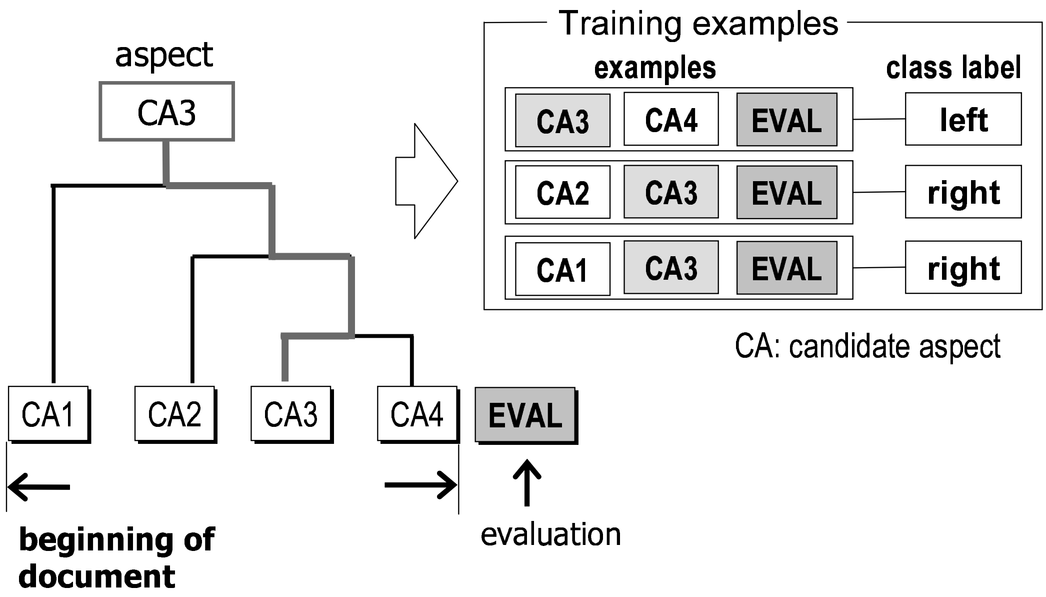


Figure 5.2. Tournament model proposed by Iida *et al.* (2003)

not and whether the pair is an opinion or not). In the opinion-hood determination step, we can use the information about whether the evaluation has a corresponding aspect or not. We therefore create two separate models for the cases where the evaluation does and does not have an explicit aspect. These models can be implemented in a totally machine learning-based fashion.

### 5.2.2 Existing techniques for anaphora resolution

Computational approaches to anaphora resolution have been roughly evolving in two different but complementary directions: theory-oriented rule-based approaches and empirical corpus-based approaches.

In rule-based approaches [Mitkov, 1997; Baldwin, 1995; Nakaiwa and Shirai, 1996], efforts have been directed to manual encoding of various linguistic cues into a set of rule based on theoretical linguistic work such as Centering Theory [Grosz *et al.*, 1995; Kameyama, 1986] and Systemic Theory [Halliday and Hasan, 1976]. The best-achieved performance for the coreference task test set of MUC-7 [2] was

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

around 70% precision with 60% recall, which is still far from being satisfactory for many practical applications. Worse still, a rule set tuned for a particular domain is unlikely to work equally for another domain due to domain-dependent properties of coreference patterns. Given these facts, further manual refinements of rule-based models will be prohibitively costly.

Corpus-based empirical approaches, such as [Soon *et al.*, 2001; Ng and Cardie, 2002; Iida *et al.*, 2003; Ng, 2004], on the other hand, are cost effective, while having achieved a better performance than the best-performing rule-based systems for the test sets of MUC-6 and MUC-7. Based on these findings, we introduce a corpus-based empirical approach to an aspect identification model.

## 5.3 Experiments

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

### 5.3.1 Training/evaluation data

To Japanese review articles in the automobile domain (4,442 sentences), we annotated evaluation and aspect tags according to the definition we described in Section 2.2. Note that our aim in this chapter is to extract aspect-evaluation pairs, therefore, we asked the annotator to choose the aspect lowest in the hierarchy when they select the aspect of the evaluation, if some aspects are in a hierarchical relation with each other. The hierarchical relation we mentioned includes part-of (e.g. "the switch of the air conditioner") and attribute-of (e.g. "the sound of the engine") relations. For example, in *"the sound of the engine is good"*, only "*sound*" is annotated as the aspect of the evaluation "*good*".

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

### 5.3.2 Experimental method

As preprocessing, we analyzed the opinion-tagged corpus using the Japanese morphological analyzer *ChaSen*[3] and the Japanese dependency structure analyzer *CaboCha* [4].

We used Support Vector Machines[5] to train the models for aspect identification, pairedness determination and opinion-hood determination. Support Vector Machines (SVMs) are binary classifiers proposed by Vapnik [1998]. SVMs have applied to various real-world applications, such as text categorization and character recognition, and have been proven successful. We used the 2nd degree polynomial kernel as the kernel function for SVMs. Evaluation was performed by 10-fold cross validation using all the data.

#### Order of model application

To examine the effects of appropriately choosing the order of model application we discussed in the previous section, we conducted four experiments using different orders:

Proc.1: opinion-hood determination → pairedness determination→ aspect-identification

Proc.2: opinion-hood determination → aspect identification → pairedness determination

Proc.3: aspect identification → opinion-hood determination → pairedness determination

Proc.4: aspect identification → pairedness determination → opinion-hood determination

Note that Proc.4 is our proposed ordering.

In addition to these models, we adopted a baseline model. In this model, if the candidate evaluation and a candidate aspect are connected via a dependency relation, the candidate evaluation is judged to have an aspect. When none of the

[3] http://chasen.naist.jp/

[4] http://chasen.org/˜taku/software/cabocha/

[5] We use a package TinySVM (http://chasen.org/˜taku/software/TinySVM/)

Table 5.1. Features used in each model. AI: the aspect identification model,PD: the pairedness determination model, OD: the opinion-hood determination model.

| | Proc.1 | | | Proc.2 | | | Proc.3 | | | Proc.4 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | AI | PD | OD | AI | PD | OD | AI | PD | OD | AI | PD | OD(A-E) | OD(E) |
| *a* | √ | √ | √ | √ | √ | √ | √ | √ | √ | √ | √ | √ | √ |
| *b* | √ | √ | | √ | | √ | √ | √ | √ | √ | √ | √ | |

candidate aspects have a dependency relation, the candidate evaluation is judged not to have an aspect.

### Dictionaries

We use dictionaries for identification of aspect and evaluation candidates. We constructed an aspect dictionary and an evaluation dictionary from review articles about automobiles (230,000 sentences in total) using the semi-automatic method described in Chapter 4.

We assume that we have large dictionary which covers most of the aspect and evaluation phrases, thus we added to the dictionaries expressions which frequently appear in the opinion-tagged corpus. The sizes of the final dictionaries become 3,777 aspect phrases and 3,950 evaluation phrases.

### Features

We extracted the following two types of features from aspect candidates and evaluation candidates:

(a) surface spelling and part-of-speech of the target evaluation expressions, as well as those of their dependent phrase and those in their depended phrases

(b) relation between the target evaluation and its aspect candidate (distance between them, existence of dependency relation, existence of a co-occurrence relation)

We extracted (b) if the model could use both the aspect and the evaluation information.

Table 5.1 summarizes which of the following types of features are used in each model. In Proc.4, we can use the information about whether the evaluation has a corresponding aspect or not for opinion-hood determination. We therefore create two separate models for when the evaluation does and does not have an aspect.

Existence of co-occurrence relations are determined by reference to a predefined co-occurrence list that contains aspect-evaluation pair information such as "*syakou* (height of vehicle) – *hikui* (low)". We created the list from the 230,000 sentences described in previous section by applying the aspect and evaluation dictionaries and extracting aspect-evaluation pairs if there is a dependency relation between the aspect and the evaluation. The number of pairs we extracted was about 48,000.

### 5.3.3 Results

Table 5.2 shows the results of opinion extraction. In the table, "evaluation with explicit aspect" indicates recall and precision of aspect-evaluation pairs where both an evaluation and its aspect appear in the text, and "evaluation without explicit aspect" indicate the result where the evaluation appears in the text while its aspect is missing. "aspect-evaluation pairs" is sum of above two rows.

We evaluated the results by recall $R$ and precision $P$ defined as follows (For simplicity, we substitute "A-E" for aspect-evaluation pair):

$$\begin{aligned} R &= \frac{\text{correctly extracted A-E opinions}}{\text{total number of A-E opinions}}, \\ P &= \frac{\text{correctly extracted A-E opinions}}{\text{total number of A-E opinions found by the system}}. \end{aligned}$$

We also use the F-measure, which is the harmonic mean of precision and recall:

$$F - measure = \frac{2 \times R \times P}{(R + P)}$$

In order to demonstrate the effectiveness of the information about the candidate aspect, we evaluated the results of pair extraction and opinion-hood determination separately. Table 5.3 and Table 5.4 show the results. In these tables, AI indicates the aspect identification model, PD indicates the pairedness determination model and OD indicates the opinion-hood determination model. In the pair

Table 5.2. The precision and the recall for opinion extraction

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

extraction, we assume that the evaluation is given, and evaluate how successfully aspect-evaluation pairs are extracted.

### 5.3.4 Discussions

From Table 5.2, we can see that recall of our model outperforms the baseline model, since this method can extract pairs which are not connected via a dependency relation in the sentence. Moreover, the precision of our method outperforms the baseline model. We also see our proposed ordering outperforms the other orderings, and gets the best F-measure.

In what follows, we discuss the results of pair extraction and opinion-hood determination.

**Pair extraction:** From Table 5.3 and Table 5.4, we can see that carrying out aspect identification before pairedness determination outperforms the reverse or-

Table 5.3. The result of pair extraction

| procedure | precision | recall |
|---|---|---|
| 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) |

Table 5.4. The result of opinion-hood determination

| procedure | precision | recall |
|---|---|---|
| OD | 74.0% (1554/2101) | 60.2% (1554/2581) |
| AI→OD | 82.2% (1709/2078) | 66.2% (1709/2581) |

dering by 11% in precision and 3% in recall. This result supports our expectation that knowledge of aspect information contributes to aspect-evaluation pair extraction. Focusing on the rows labeled "(dependency)" and "(no dependency)" in Table 5.3, while 80% of the aspect-evaluation 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 aspect and evaluation are not in a direct dependency relation.

According to our error analysis, a major source of errors lies in the aspect identification task. In this experiment, the precision of aspect identification is 78%. A major reason for this problem was the coverage of the dictionary. In addition, the system causes a false decision the aspect appears in the preceding sentence.

**Opinion-hood determination:** Table 5.4 shows that carrying out aspect identification followed by opinion-hood determination outperforms the reverse ordering, which supports our expectation that knowing the aspect information helps opinion-hood 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 address the issue of modifying the feature set to make the model more sensitive to modality-oriented distinctions such as subjunctive and conditional expressions.

### 5.3.5 Subject detection

As mentioned in Section 5.1, we have so far put aside the task of filling the *Subject* slot assuming that it is not a bottle-neck problem. Here, we provide a piece of evidence for this assumption by briefly reporting on the results of another experiment.

For the experiment, we created a corpus annotated with subject-evaluation pairs. The corpus consisted of 308 weblog articles in the automobile domain (3,037 sentences) containing 870 subject-evaluation pairs.

We assumed that for each given article, all the subject expressions and evaluation expressions had been properly identified. The task was to identify the subject corresponding to a given evaluation expression. For this task, we implemented simple heuristics as follows:

1. If there are any subject expressions preceding the given evaluation expressions, choose the nearest one to the evaluation

2. Otherwise, choose the first one of those following the evaluation expression

The precision was 0.92 (822/890), and the recall was 0.94 (822/870). A major error was that the heuristics could not appropriately handle opinions that exhibited a comparison between a subject and its counterpart. However, this problem was not a big deal in terms of frequency. The results suggest that the problem of identifying subject-evaluation pairs is solvable with reasonably high precision and recall provided that subject expressions are properly identified. Subject expression identification is a subclass of named entity recognition, which has been actively studied for a decade. We are planning to incorporate state-of-the-art techniques for named entity recognition to the overall opinion mining system we are new developing.

## 5.4 Summary

In this chapter, we focused on the aspect–evaluation pair extraction, proposed a machine learning-based method consisting of three components: aspect identification, aspect-evaluation pairedness determination and opinion-hood determination. We evaluated the method in review articles, and showed that identifying the corresponding aspect for a given evaluation expression is effective in both aspect-evaluation pairedness determination and opinion-hood determination.

We have so far considered the approach relies on the dictionaries in detecting evaluation and aspect candidates. However, the result showed that the coverage of the aspect dictionary is a bottleneck of the approach, we therefore have explored an approach which does not use the aspect dictionary. At the next chapter, we discuss the method to extract aspect-evaluation pairs and aspect-of relations without using the aspect dictionary.

# Chapter 6

# Extracting aspect-evaluation and aspect-of relations

## 6.1 Introduction

In the previous section, we discussed the method for extracting aspect-evaluation pair using domain-specific dictionary. However, as we describe below, aspect expressions are tend to be heavily domain-dependent, and it is not easy to create an exhaustive list of aspects.

In this chapter, we discuss the method for aspect-evaluation relation extraction without relying on an aspect dictionary. Furthermore, we consider the task of extracting hierarchical relations between aspects.

## 6.2 Resource availability

Before designing a model for our opinion extraction task, it is important to note that aspect phrases are open-class words and tend to be heavily domain-dependent. In fact, according to our investigation on our opinion-annotated corpus, the number of aspect types is nearly 3,200, and we found only 3% of all aspect expressions appeared in two or more domains as shown in Figure 6.1. Given this, it is not realistic to assume the availability of any list of aspect expressions applicable to a wide range of domains with a broad coverage. One

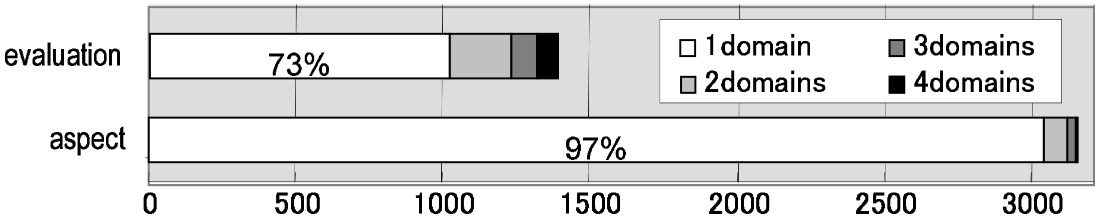


Figure 6.1. Numbers of the expressions used over four domains

important issue, therefore, is how to identify aspects without any predefined list of candidate aspect expressions.

For evaluation phrases, on the other hand, the number of types is nearly 1400, and 28% of all evaluation expressions appear in multiple domains. This indicates that evaluations are more likely to be used commonly across different domains compared with aspects. To prove this assumption, we actually constructed a dictionary of evaluation expressions from automobile reviews (230,000 sentences in total) using the semi-automatic method proposed in Chapter 4. We expanded the dictionary to include entities by hand from external resources such as publically available ordinal thesauri. As a result, we collected 5,550 entries, which is now available from http://cl.naist.jp/~nozomi-k/evaluative_expressions.html. According to our investigation of the coverage for the dictionary, approximately 90% on average (at least 80%) of the evaluations annotated in our opinion-annotated corpus are covered by the dictionary. This result supports our assumption about the availability of an open-domain lexicon of evaluation expression. In our experiments, we used this evaluation dictionary.

Given these considerations about the resource availability, we design the process of extracting ⟨Subject, Aspect, Evaluation⟩ as follows:

1. **Aspect-evaluation relation extraction**: For each of the candidate evaluation that are selected from a given document by dictionary look-up, identify the target of the evaluation. In this step, we use the evaluation dictionary mentioned above. Here the identified target may be an aspect of an subject (e.g. *the quality (is amazing)*) but may also be a subject itself (e.g. *Canon G3 (is well-designed)*. Hereafter, we use the term *aspect* to refer to both an aspect of a subject and a subject itself, since the subject can be regarded

as the top element in the hierarchical chain of aspects.

2. **Opinion-hood determination**: Judge whether the obtained pair ⟨aspect, evaluation⟩ is an expression of an opinion or not by considering the given context. If it is, go to step 3; otherwise, return to step 1 with a new candidate evaluation expression.

3. **Aspect-of relation extraction**: If the identified aspect is not an opinion subject, search for its parent, i.e. the target whose part or attribute is the current aspect. Repeat step 3 until reaching an opinion subject or no parent is found.

## 6.3 Related work on opinion extraction

As we mentioned in Chapter 3, approaches to the aspect-evaluation extraction task mainly use simple proximity- or pattern-based techniques. For example, Tateishi *et al.* [2004] implemented five syntactic patterns and Popescu and Etzioni [2005] used ten syntactic patterns.

Such an approach is limited in two respects. First, it assumes the availability of a list of potential aspect expressions as well as evaluation expressions; however creating such lists for a variety of domains can be expensive because of the domain dependency of aspect expressions. In contrast, our method does not require any aspect lexicon. Second, their approach lacks the perspective of viewing aspect-evaluation extraction as a specific type of predicate-argument structure analysis, i.e. the task of identifying the arguments of a given predicate in a given text, and fails to benefit from the state-of-the-art of this rapidly growing field. An exception is the model reported by Kanayama and Nasukawa [2004], which uses a component of an existing machine translation system to identify the "aspect" argument of a given "evaluation" predicate. However, the MT component they use is not publicly available and, even if it is, will be difficult to adapt it to tasks in hand due of the opaqueness of its mechanism. Our approach aims to develop a more generally applicable model of aspect-evaluation extraction.

Aspect-of relations can be regarded as a subtype of bridging reference [Clark, 1977], also known as indirect anaphora or associative anaphora. Bridging reference is the referent of a definite description implicitly related to some previously mentioned entity. For example, we can see a relation of bridging reference between "*the door*" and "*the room*" in the sentences "*She entered the room. The door closed automatically.*". A common approach is to use co-occurrence statistics between the referring expression (e.g. "*the door*" in the above example) and the mentioned entity (e.g. "*the room*") [Bunescu, 2003; Poesio *et al.*, 2004]. Bunescu [2003] and Poesio *et al.* [2004] use the number of web pages which contain both the referring expression and the mentioned entity being queried as a measure of the strength of association. Our approach newly incorporates automatically induced syntactic patterns as contextual clues into such a co-occurrence statistics-based model, producing significant improvements of accuracy.

In the current relation extraction task, approaches based on kernel methods achieve the best performance [Zelenko *et al.*, 2003; Culotta and Sorensen, 2004, etc.]. Kernel methods are the techniques which compute a kernel function to measure the similarity between data instances. Several kernel functions were tried for this task: Culotta and Sorensen extended the work by using dependency tree kernel which represent the syntactic relations between the words of a sentence. Harabagiu *et al.* [2005] introduced the semantic resources to the approach based on dependency tree kernel. These researchers have been exploring techniques for extracting relations between two entities which are already identified. In the opinion extraction task, on the other hand, aspects are heavily domain dependent, which indicates that it is difficult to assume that aspects are already identified. In this chapter, we explore a method for extracting relations and recognizing aspects simultaneously focusing on the task of extracting aspect-of and aspect-evaluation relations.

## 6.4 Method for opinion extraction

### 6.4.1 Our approach

The key idea for our relation extraction subtasks is to combine the following two kinds of information using a machine-learning technique.

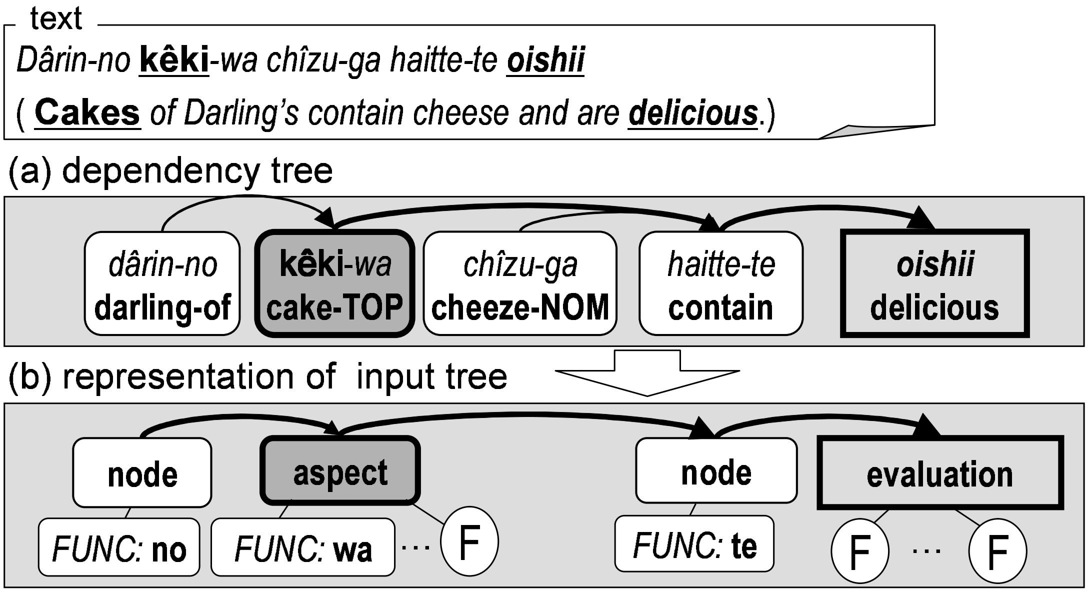


Figure 6.2. Representation of input data

- Contextual clues: Syntactic patterns such as

  | | | |
  |---|---|---|
  | ⟨*Aspect*⟩*-ga* | *X-te,* | ⟨*Evaluation*⟩ |
  | ⟨*Aspect*⟩-NOM | *X*-CONJ | ⟨*Evaluation*⟩ |

  which matches such a sentence as

  ⟨*sekkyaku*⟩*-ga* *kunrens-aretei-te* ⟨*kimochiyoi*⟩
  ⟨*service*⟩-NOM *be trained*-CONJ ⟨*feel comfortable*⟩
  (*The waiters were trained, so I felt comfortable.*)

  are considered to be useful for extracting relations between slot fillers when they appear in a single sentence (Here, ⟨⟩ indicates a slot filler). We employ a supervised learning technique to search for useful contextual clues.

- Context-independent statistical clues: Some examples are the statistics of aspect-aspect and aspect-evaluation co-occurrences, which are extracted to be useful clues. We obtain such statistical clues automatically from a large collection of raw documents.

### 6.4.2 Supervised learning of contextual clues

Let us consider the problem of searching for the aspect of a given target evaluation expression $t$, which can be decomposed into binary classification problems of deciding whether each pair of candidate aspect $c$ and target $t$ is in an aspect-evaluation relation or not. Our goal is to learn a discrimination function for this classification problem. With such a function is obtained, we can identify the most likely candidate aspect simply by selecting the best scored $c$-$t$ pair and, if its score is negative, conclude that $t$ has no corresponding aspect in the candidate set.

To use syntactic patterns as contextual clues, we represent each $c$-$t$ pair as such a tree as illustrated in Figure 6.2 if $c$ and $t$ appear in the same sentence. Among various classifier induction algorithms for tree-structured data, in our experiments, we have so far examined Kudo and Matsumoto [2004]'s algorithm, which is implemented as the package *BACT*. Given a set of training examples represented as ordered trees labeled either positive or negative, this algorithm learns a list of weighted decision stumps as a discrimination function with a Boosting algorithm . Each decision stump is associated with tuple $\langle s, l, w\rangle$, where $s$ is a subtree appearing in the training set, $l$ a label, and $w$ a weight, indicating that if a given input includes $s$, it gives $w$ votes to $l$. The strength of this algorithm is that it deals with structured features and allows us to analyze the utility of features.

Each $c$-$t$ pair is encoded as a tree in the following manner. First, we use a dependency parser to obtain the dependency parse tree as in Figure 6.2 (a), where "*kêki* (*cake*)" is assumed to be $c$ and "*oishii* (*delicious*)" $t$. Next, we extract the path from $t$ to $c$ together with the daughter nodes of $t$ and $c$ as in Figure 6.2 (b), where the node corresponding to "*dârin-no* (*Darling's-of*)" is remained because it is a daughter of $c$, i.e. "*kêki-wa* (*cake*-TOP)". The information of content words is then replaced with node types, either *evaluation*, *aspect* or *node*, to avoid inducing domain-specific patterns, while keeping the information of function words as in, for example, the node labeled "FUNC:no" in Figure 6.2 (b). Besides such function word nodes, we add extra nodes, depicted by the ones labeled "*F*" in the figure, that represent the features summarized in Tables 6.1, 6.2 and 6.3.

Note that the same story holds for aspect-of relation extraction as well if we replace the "evaluation" above with "aspect". We use the same algorithm for aspect-of relations in an analogous manner.

### 6.4.3 Context-independent statistical clues

We also introduce following three kinds of statistical clues to adapt the model for a given domain.

**i. Aspect-evaluation/aspect-aspect co-occurrences**

Among various ways to estimate co-occurrence (e.g. the number of hits returned from a search engine), in the experiments we report below, we counted aspect-aspect and aspect-evaluation co-occurrences in 1.7 million weblog posts using the patterns

- "⟨aspect⟩ *ga/wa/mo* ⟨evaluation⟩
  (⟨aspect⟩ *is* ⟨evaluation⟩)"
- "⟨aspect_A⟩ *no* ⟨aspect_B⟩ ga/wa"
  (⟨aspect_B⟩ *of* ⟨aspect_A⟩ *is*)"

To avoid the data sparseness problem, we use the Probabilistic Latent Semantic Indexing (PLSI) [Hofmann, 1999]. We can calculate the joint probability $P(A, B)$ even if $A$ and $B$ do not directly co-occur, since PLSI assumes a set of latent class of co-occurrence:

$$P(A, B) = \sum_{z \in Z} P(A|z)P(B|z)P(z)$$

where $Z$ denotes a set of latent class of co-occurrence. We can calculate pointwise mutual information, conditional probabilities, etc. from the estimated distribution $P(A, B)$. In our experiment, we use conditional probabilities $P(Aspect|Evaluation)$ and $P(Aspect\_A|Aspect\_B)$, and pointwise mutual information $PMI(Aspect, Evaluation)$ and $P(Aspect\_A, Aspect\_B)$ as described below. We then incorporate the information of these probability scores into the learning model described in Section 6.4.2 by encoding them as a feature that indicates the relative score rank of each candidate in a given candidate set (see Tables 6.1, 6.2 and 6.3).

**ii. Aspect-hood of candidate aspects**

Aspect-hood is an index of the degree to which the term is used as an aspect within a given domain. First, we extract the expression $X$ which appear in the form "*Subject no X* (*X of Subject*)", and then extract the expression $Y$ which appear in the form "*X no Y* (*Y of X*)". We calculate the aspect-hood of the expressions $X$ and $Y$ as pointwise mutual information [Church and Hanks, 1989]:

$$PMI(X,Y) \quad = \quad log_2 \frac{N \times count(X,Y)}{count(Y) \times count(X)}$$

where $count(X,Y)$ is the number of times $X$ occurred in $Y$, and $N$ is the total number of times all pairs occurred in the form "$X$ no $Y$". $count(X)$ (or $count(Y)$) is the number of $X$ (or $Y$) occurred. This score is also used as a features (see Tables 6.1, 6.2 and 6.3).

**iii. Statistical inference of aspect-aspect relation classes**

Aspect-aspect co-occurrences are good clues for extracting aspect-of relations. However, many other types of relations can hold between two nouns which appear in "*A no B* (*B of A*)" form. It is not clear whether the two nouns have aspect-of relation or not. For example, "*watashi no kuruma* (*my car*)" and "*kuro no seifuku* (*the black uniform*)" appear in the form of "*A no B*", however, the former relation indicates possession and the latter represents a property (color) of the uniform, that is, not a part-of or attribute-of relation.

For this problem, we create the model to estimate the aspect-of relation using the maximum entropy model [Berger *et al.*, 1996]. The maximum entropy method has been successfully applied in many tasks in natural language processing such as part-of-speech tagging. This method is the algorithm to estimate the conditional probability $p(y|x)$ from the training examples $(x_i, y_i)$. In our problem, $x$ represents a pair of nouns, and $y$ represents a relation (aspect-of or other). In maximum entropy estimation, we define a set of feature function $F = f_i(x,y)$ to model an example $(x,y)$, and model the conditional probability $p(y|x)$ as follows:

$$p(y|x) = \frac{1}{Z(x)} exp(\sum_i \lambda_i f_i(x,y))$$

$$Z(x) = \sum_y exp(\sum_i \lambda_i f_i(x,y)$$

where $\lambda_i$ is a weight parameter for the feature function $f_i(x,y)$. In our experiment, we used MaxEnt package which is available from http://maxent.sourceforge.net/.

We created labeled data, which consists of pair of nouns, annotated with 'aspect-of' and 'other' relation tags, and learned the model with the features: verbs or adjectives co-occurred with $A$ or $B$ and the semantic classes of $A$ or $B$ derived from the Japanese thesaurus "Nihongo Goi Taikei" [Ikehara *et al.*, 1997]. The size of the labeled data is nearly 5,300, half of the data is "aspect-of" and the remains is "other".

We estimate the label of a pair of candidates and aspects using this model, then encode the label and its probability as a features (see Table 6.2).

### 6.4.4 Extraction of aspect-evaluation relations

Syntactic pattern induction as described in Section 6.4.2 can apply only when an aspect-evaluation relation appears in a single sentence. We therefore build a separate model for inter-sentential relation extraction, which is carried out after intra-sentential relation extraction.

1) Intra-sentential relation identification: Given a target evaluation, select the most likely candidate aspect $c^*$ within the evaluation sentence with the intra-sentential model described in 3.2.1. If the score of $c^*$ is positive, return $c^*$; otherwise, go to the inter-sentential relation extraction phase.

2) Inter-sentential relation identification: Search the most likely candidate aspect for the sentences preceding the one of the target evaluation. This task can be regarded as a zero-anaphora resolution problem. For this purpose, we employ the tournament model which is a supervised learning model for zero-anaphora resolution proposed by [Iida *et al.*, 2003].

The specific features we used in the experiments are summarized in Tables 6.1 and 6.2.

### 6.4.5 Opinion-hood determination

Evaluation phrases do not always constitute opinion units in a given domain. Consider an example from the digital camera domain, "*The weather was good, so I went to the park to take some pictures of roses*". The evaluation phrase "*good*" expresses the evaluation for "*the weather*", but "*the weather*" is not an aspect of digital cameras. Therefore, "the weather–good" is not an opinion which we aim to extract. We can consider that the task of judging whether the obtained opinion unit is a real opinion or not in a given domain is a binary classification task. We introduce the opinion-hood determination model learned by Support Vector machines. The specific features we used in the experiments are summarized in Table 6.3.

### 6.4.6 Extraction of aspect-of relations

We also approach the aspect-of relation extraction by decomposing it into two subtasks (explained in Section 6.4.4), and build a separate component proposed in aspect-evaluation relation extraction problem.

## 6.5 Experiments

We conducted experiments with our Japanese opinion-annotated corpus to empirically evaluate the performance of our approach. In these experiments, we separately evaluated the models of aspect-evaluation relation extraction, opinion-hood determination, and aspect-of relation extraction.

### 6.5.1 Common settings

We chose 395 weblog posts in the restaurant domain from our opinion-tagged corpus we described in Chapter 2. As preprocessing, we analyzed the opinion-tagged corpus using the Japanese morphological analyzer *ChaSen*[1] and the Japanese dependency structure analyzer *CaboCha* [2].

For the classifier, we used *BACT*[3] for the the intra-sentential models, and Support Vector Machines with 2nd order polynomial kernel for the inter-sentential, and opinion-hood determination models.

[1] http://chasen.naist.jp/

[2] http://chasen.org/˜taku/software/cabocha/

[3] http://chasen.org/˜taku/software/bact/

Table 6.1. Features for aspect-evaluation: $e$ denotes the evaluation and $c$ denotes the candidate

| Intra-sentential | |
|---|---|
| Feature type | Description |
| Grammatical | • Part-of-speech of $e$ and $c$ followed by IPADIC [Asahara and Matusmoto, 2003]<br>• Particle which follows $c$, such as ‘ ga (subject) ’, ‘ o (object) ’ |
| Lexical | • Whether $c$ appears in a quoted sentence<br>• Character type of $c$ (*katakana*, *English alphabet*, etc.)<br>• Suffix of $c$ (*-sei*, *-sa* (-ty), etc.) |
| Positional | • Whether $c$ precedes $e$<br>• Whether $e$ precedes $c$<br>• Position of $c$ / $e$ in the sentence (begining, end, other) |
| Syntactical | • Whether $c$ and $e$ has a immediate dependency relation |
| Semantic | • Semantic class of $c$ derived from Nihongo Goi Taikei[Ikehara *et al.*, 1997] |
| Statistical | • Co-occurrence score rank of $c$ (1st, 2nd, 3rd, 4th, other)<br>• Aspect-hood score rank of $c$ (1st, 2nd, 3rd, 4th, other) |

| Inter-sentential | |
|---|---|
| Feature type | Description |
| Grammatical | • Part-of-speech of $e$ and $c$ followed by IPADIC [Asahara and Matusmoto, 2003] |
| Lexical | • Whether $c$ appears in a quoted sentence<br>• Character type of $c$ (*katakana*, *English alphabet*, etc.)<br>• Suffix of $c$ (*-sei*, *-sa* (-ty), etc.) |
| Positional | • Sentence distance between $c$ and $e$ (1, 2, 3, 4, other)<br>• Position of $c$ / $e$ in the sentence (begining, end, other) |
| Semantic | • Semantic class of $c$ derived from Nihongo Goi Taikei[Ikehara *et al.*, 1997] |
| Statistical | • Which of two candidates have high co-occurrence score<br>• Which of two candidates have high aspect-hood score |

Table 6.2. Features for aspect-of: $a$ denotes the aspect and $c$ denotes the candidate

| Intra-sentential | |
|---|---|
| Feature type | Description |
| Grammatical | • Part-of-speech of $a$ and $c$ followed by IPADIC [Asahara and Matusmoto, 2003]<br>• Particle which follows $c$, such as ‘ ga (subject) ’, ‘ o (object) ’ |
| Lexical | • Whether $c$ appears in a quoted sentence |
| Positional | • Whether $c$ precedes $a$<br>• Whether $a$ precedes $c$<br>• Position of $c$ / $a$ in the sentence (begining, end, other) |
| Syntactical | • Whether $c$ and $a$ has a immediate dependency relation |
| Semantic | • Semantic class of $c$ derived from Nihongo Goi Taikei[Ikehara *et al.*, 1997] |
| Statistical | • Co-occurrence score rank of $c$ (1st, 2nd, 3rd, 4th, other)<br>• Aspect-hood score rank of $c$ (1st, 2nd, 3rd, 4th, other)<br>• Relation label between $c$ and $a$ estimated by statistical inference model |

| Inter-sentential | |
|---|---|
| Feature type | Description |
| Grammatical | • Part-of-speech of $a$ and $c$ followed by IPADIC [Asahara and Matusmoto, 2003] |
| Lexical | • Whether $c$ appears in a quoted sentence<br>• Character type of $c$ (*katakana*, *English alphabet*, etc.)<br>• Suffix of $c$ (*-sei*, *-sa* (-ty), etc.) |
| Positional | • Sentence distance between $c$ and $a$ (1, 2, 3, 4, other)<br>• Position of $c$ / $a$ in the sentence (begining, end, other) |
| Semantic | • Semantic class of $c$ derived from Nihongo Goi Taikei[Ikehara *et al.*, 1997] |
| Statistical | • Which of two candidates have high co-occurrence score<br>• Which of two candidates have high aspect-hood score<br>• Relation label between $c$ and $a$ estimated by statistical inference model |

Table 6.3. Features for opinion-hood determination: $e$ denotes the evaluation and $c$ denotes the candidate

| Intra-sentential | |
|---|---|
| Feature type | Description |
| Grammatical | • Part-of-speech of $e$ and $c$ followed by IPADIC [Asahara and Matusmoto, 2003]<br>• Particle which follows $c$, such as ‘ ga (subject) ’, ‘ o (object) ’ |
| Lexical | • Character type of $c$ (*katakana*, *English alphabet*, etc.)<br>• Suffix of $c$ (*-sei*, *-sa* (-ty), etc.) |
| Positional | • Position of $c$ / $e$ in the sentence (begining, end, other)<br>• Sentence distance between $c$ and $e$ (1, 2, 3, 4, other) |
| Syntactical | • Whether $c$ and $e$ has a immediate dependency relation |
| Semantic | • Semantic class of $c$ derived from Nihongo Goi Taikei[Ikehara *et al.*, 1997] |
| Statistical | • Co-occurrence score rank of $c$ (1st, 2nd, 3rd, 4th, other)<br>• Aspect-hood score rank of $c$ (1st, 2nd, 3rd, 4th, other) |

### 6.5.2 Feature lists

We summarize the features used for train and test the models in Tables 6.1 and 6.2. For opinion-hood determination, we used the some kind of features shown in Table 6.3.

### 6.5.3 Models

The results are summarized in Table 6.4, where five models are compared for each of the two subtasks: aspect-evaluation relation extraction and aspect-of relation extraction. The following is a summary of each model for the former subtask:

**Baseline A-E model** simulates the algorithm proposed by [Tateishi *et al.*, 2004]:

1. If there are any candidate aspects which match the following extraction patterns:
   - ⟨Aspect⟩ *ga/wa/mo/no/ni/wo/de* ⟨Evaluation⟩
   - ⟨Evaluation⟩ syntactically depends ⟨Aspect⟩

choose the nearest one as the aspect of the evaluation

2. Otherwise, choose the candidate aspect with the highest aspect-evaluation co-occurrence score.

**Context-only A-E model:** uses contextual pattern-based clues (6.4.2) but not statistical clues (6.4.3) and works in the manner as described in 6.4.4.

**Proposed A-E model:** uses both contextual and statistical clues together by encoding the aspect-evaluation co-occurrence score and the aspect-hood score (6.4.3.i) as a set of additional features to the tree-representation (Figure 6.2) of a given input.

**Proposed-MI A-E model:** uses the same clues as the proposed A-E model except that it uses point-wise mutual information as the aspect-evaluation co-occurrence score instead of the conditional probabilities of aspect-evaluation co-occurrence, which is used in the proposed model.

**Proposed-dic A-E model:** incorporates an aspect expression dictionary in the Proposed A-E model instead of automatically calculated aspect-hood scores. The aspect expression dictionary was manually created for the restaurant domain containing 6,129 expressions.

Comparing the Baseline model with the Context-only model shows the effects of the supervised learning of contextual pattern features, while a comparison of the Context-only and Proposed models shows the joint effects of combining contextual and statistical clues. The performance of the Proposed-dic model provides an estimation of the upper-bound of the improvements that could be gained by accurate estimation of the aspect-hood of each candidate aspect.

The Baseline model (the Baseline Aspect-of model) we implemented for aspect-of relation extraction relies only on the aspect-aspect co-occurrence score, which simulates the method for bridging reference resolution proposed by [Bunescu, 2003]:

1. Select the expression which has highest scores of pointwise mutual information, if there are any candidate in the sentence which the aspect appear.

2. Otherwise, choose the nearest one which co-occur with the aspect.

The other four models for aspect-of relation extraction were created analogously to the above A-E models. The Proposed Aspect-of model uses the information of the statistically estimated aspect-aspect relation class for each candidate aspect in addition to the aspect-evaluation co-occurrence score and the aspect-hood score.

### 6.5.4 Evaluation

We conducted 5 fold cross validation using all the data, and evaluated the results by recall $R$ and precision $P$ defined as follows

$$\begin{aligned} R &= \frac{\text{correctly extracted relations}}{\text{total number of relations}}, \\ P &= \frac{\text{correctly extracted relations}}{\text{total number of relations found by the system}}. \end{aligned}$$

### 6.5.5 Results and discussions

Table 6.4 shows the result of aspect-evaluation and aspect-of relation extraction tasks, and opinion-hood determination. As for the aspect-evaluation relation extraction, concerning the intra-sentential cases, we can see that the models using the contextual clues show nearly 10% improvement in both precision and recall.

This indicates that the machine learning-based method devised for predicate-argument structure analysis improves the performance of aspect-evaluation relation extraction. Similar results are obtained in aspect-of relation extraction. The models using the contextual clues achieved more than 10% improvement in precision and 20% improvement in recall over the co-occurrence statistics-based model. We can say that contextual clues are also useful in aspect-of relation extraction.

Table 6.5 compares the difference in the cases where the candidate expressions in aspect-evaluation or aspect-of relation are syntactically dependent. "A → E(B)" indicates that A is dependent on E(valuation) or B (e.g. "⟨サービス⟩が⟨良い⟩ (⟨*the service*⟩ *is* ⟨*good*⟩)" and "⟨インテリア⟩の⟨デザイン⟩ (⟨*design*⟩ *of* ⟨*interior*⟩)"), and "A ← E(B)" indicates that E or B is dependent on A (e.g. "⟨良い⟩⟨サービス⟩ (⟨*good*⟩⟨*service*⟩" and "⟨(素敵な)⟨デザイン⟩の⟨インテリア⟩⟩((excellent)⟨*design*⟩ *of* ⟨*interior*⟩)"). The column "no dependency" means that they are syntactically non-dependent. While it is quite natural that the precision

Table 6.4. The results of aspect-evaluation (A-E) relation, aspect-of relation and opinion-hood determination

| | | | intra-sentential | | inter-sentential | | total | |
|---|---|---|---|---|---|---|---|---|
| A-E | Baseline | precision | 0.56 | (432/774) | 0.08 | (20/235) | 0.45 | (452/1009) |
| | | recall | 0.53 | (432/809) | 0.07 | (20/274) | 0.42 | (452/1083) |
| | Context-only | precision | 0.70 | (504/723) | 0.13 | (46/360) | 0.51 | (550/1083) |
| | | recall | 0.62 | (504/809) | 0.17 | (46/274) | 0.51 | (550/1083) |
| | Proposed | precision | 0.72 | (502/694) | 0.14 | (53/389) | 0.51 | (555/1083) |
| | | recall | 0.62 | (502/809) | 0.19 | (53/274) | 0.51 | (555/1083) |
| | Proposed-MI | precision | 0.70 | (505/682) | 0.14 | (55/401) | 0.51 | (560/1083) |
| | | recall | 0.62 | (505/809) | 0.20 | (55/274) | 0.51 | (560/1083) |
| | Proposed-dic | precision | 0.80 | (482/600) | 0.17 | (83/477) | 0.52 | (565/1083) |
| | | recall | 0.60 | (482/809) | 0.30 | (83/274) | 0.52 | (565/1083) |
| aspect-of | Baseline | precision | 0.25 | (79/312) | 0.21 | (79/370) | 0.23 | (158/682) |
| | | recall | 0.34 | (79/234) | 0.10 | (79/814) | 0.15 | (158/1048) |
| | Context-only | precision | 0.41 | (122/297) | 0.30 | (222/748) | 0.33 | (344/1045) |
| | | recall | 0.52 | (122/234) | 0.27 | (222/814) | 0.33 | (344/1048) |
| | Proposed | precision | 0.43 | (139/321) | 0.34 | (247/814) | 0.37 | (386/1045) |
| | | recall | 0.59 | (139/234) | 0.30 | (247/814) | 0.37 | (386/1048) |
| | Proposed-MI | precision | 0.38 | (147/387) | 0.33 | (222/660) | 0.35 | (369/1047) |
| | | recall | 0.62 | (147/234) | 0.27 | (222/814) | 0.35 | (369/1048) |
| | Proposed-dic | precision | 0.52 | (145/281) | 0.42 | (319/761) | 0.45 | (464/1042) |
| | | recall | 0.62 | (145/234) | 0.39 | (319/814) | 0.44 | (464/1048) |
| | opinion-hood | precision | | | | | 0.51 | (488/949) |
| | | recall | | | | | 0.45 | (488/1083) |

Table 6.5. Results of Intra-sentential cases

| | A → E(B) | | A ← E(B) | | no dependency | |
|---|---|---|---|---|---|---|
| Context-only A-E | 0.84 | (282/337) | 0.66 | (109/164) | 0.37 | (113/308) |
| Proposed A-E | 0.81 | (273/337) | 0.68 | (112/164) | 0.38 | (117/308) |
| Context-only aspect-of | 0.72 | (67/93) | 0.2 | (1/5) | 0.4 | (54/136) |
| Proposed aspect-of | 0.73 | (68/93) | 0.4 | (2/5) | 0.51 | (69/136) |

is much higher when A is dependent on E (or B), our models achieves about 40% precision in the cases where the candidates have no syntactic relation with the other.

Among four non-baseline models in the aspect-evaluation relation extraction of Table 6.4, no significant improvement was observed. However, concerning the inter-sentential cases, we can see constant improvement according to the quantity of information used in the models, showing that the context-independent information of the candidate and co-occurrence statistics are important clues for finding the aspect expressions appearing beyond sentence boundaries.

For the aspect-of relation extraction, on the other hand, there is significant improvement in the Proposed model compared with the Context-only model. As far as the experiments show, the point-wise mutual information score does not give better performance than that the conditional probability score for co-occurrence measurement. Although there is still much room for improvement, the notable difference between the Proposed and Proposed-dic models shows that accurate estimation of the aspect-hood of candidate aspect expressions has a potential effect.

One of the reasons of low performance of aspect-of relation extraction is that the evaluation criteria is a bit too strict. The extracted aspect-aspect and subject-aspect relations are evaluated against the human annotated gold-standard in a strict manner. For example, when the gold-standard data includes the chain of aspect-of relations A-B and B-C, and the system extracts aspect-of relation A-C, it is evaluated as incorrect. In some application domains this kind of skipping aspect-of relation may not raise a severe issue. If we assume that A-C is also correct, the Proposed models achieve nearly 10% improvement in both recall and precision as shown in Table 6.6.

Table 6.6. Result of spect-of relation extraction (allow for skipping case)

| | precision | | recall | |
|---|---|---|---|---|
| Baseline | 0.27 | (175/682) | 0.17 | (175/1048) |
| Context-only | 0.44 | (458/1047) | 0.44 | (458/1048) |
| Proposed | 0.45 | (474/1047) | 0.45 | (474/1048) |
| Proposed-MI | 0.49 | (510/1047) | 0.49 | (510/1048) |
| Proposed-dic | 0.55 | (573/1047) | 0.55 | (573/1048) |

It is also crucial to address the problem of inter-sentential cases of relation extraction. For this problem, we have so far simply applied an existing machine learning-based model for zero-anaphora resolution [Iida *et al.*, 2003]. Given the results shown in Table 6.4, however, it is clear that this model needs considerable refinements to adapt to our task, which include reconsideration of the way of incorporating statistical clues into supervised learning.

Opinion-hood determination also posts a challenging problem. For example, sentence (1) includes the writer's evaluation on the shrimps served at a particular restaurant. In contrast, very similar sentence (2) does not constitute evaluation since it is a generic description of the writer's taste. The wording is, however, so similar that our models have difficulty in learning the difference.

(1) *watashi-wa konomise-no ebi-ga suki-desu*
*I*-TOP *the restaurant shrimp*-NOM *like*
(*I like the shrimps of the restaurant.*)

(2) *watashi-wa ebi-ga suki-desu*
*I*-TOP *shrimp*-TOP *like*
(*I like shrimp.*)

Thus we need to conduct further investigation in order to resolve this kind of problems.

### 6.5.6 Portability of intra-sentential model

We next evaluate effectiveness of the contextual clues leaned in a domain to other domains by testing a model trained on a certain domain to other domains. We

Table 6.7. Comparing intra-sentential models among three domains

| | test | | restaurant | cellular phone | automobile |
|---|---|---|---|---|---|
| A-E | same domain | P | 0.72 (502/694) | 0.75 (522/693) | 0.76 (562/738) |
| | | R | 0.62 (502/809) | 0.63 (522/833) | 0.65 (562/870) |
| | other domains | P | 0.75 (485/646) | 0.76 (527/698) | 0.73 (541/742) |
| | | R | 0.60 (485/809) | 0.63 (527/833) | 0.62 (541/870) |
| aspect-of | same domain | P | 0.43 (139/321) | 0.62 (139/224) | 0.66 (185/280) |
| | | R | 0.59 (139/234) | 0.60 (139/230) | 0.66 (185/279) |
| | other domains | P | 0.55 (134/245) | 0.60 (138/230) | 0.62 (201/323) |
| | | R | 0.57 (134/234) | 0.60 (138/230) | 0.72 (201/279) |

selected two new domains, cellular phone and automobile, and added 290 weblog posts for each of them. The results are evaluated by precision $P$ and recall $R$ which we defined above.

We now have three models in different domains and applied the models to analyze weblog posts in other domains. We trained a model on two domains, then tested it on remaining domain. Table 6.7 shows the results of the experiment. Compared with the model trained on the same domain, we can see that the model trained on different domains shows comparable results. This indicates that the contextual clues learned in a domain are effective in another domain, showing the portability of our intra-sentential model.

## 6.6 Summary

In this chapter, we identified the task of opinion extraction as relation extraction tasks and applied machine learning-based methods which use contextual clues and statistical clues. Our experimental results showed that the model using contextual clues improves the performance of both aspect-evaluation and aspect-of relation extraction. We also showed domain portability of the contextual clues.

# Chapter 7

# Conclusion

## 7.1 Summary

This dissertation focuses on extraction of opinions from web documents. We present four pieces of work on this topic.

As the first piece of work, we discussed the task of structuring opinions, and introduced opinion units consisting of four constituents: ⟨Opinion holder, Subject, Aspect, Evaluation⟩. We then set up an opinion extraction task based on our corpus study.

The second piece of work used particular co-occurrence patterns of evaluated subjects, focused aspects, and their evaluations to collect aspect/evaluation expressions. We reported experimental results which showed that our semi-automatic method was able to collect aspect and evaluation expressions much more efficiently than manual collection and that the co-occurrence patterns we used in the experiments worked well across two different domains.

In the third piece of work, we proposed a machine learning-based method for the extraction of opinions of consumer products by reducing the problem to that of extracting aspect-evaluation pairs from texts. The experimental results showed that identifying the corresponding aspect for a given evaluation expression is effective in both pairedness determination and opinion-hood determination.

In the final piece of work, we identified the task of opinion extraction as a relation extraction task and proposed a machine learning-based method which does not use any domain-specific aspect dictionary. Though our experimental results reflected the difficulty of the opinion extraction task, our models outperformed the baseline models in both recall and precision. We also showed that the contextual clues learned from a given domain are effective in another domain.

## 7.2 Future work

We conclude this dissertation with a discussion of future work.

### Morphological analysis for web documents

Unknown word identification is an important task for processing web documents. Until now, the main morphological analysers and dependency parsers for Japanese have been trained on “well-written” news articles, and achieve practical performance levels on news articles or other clean texts. However, web documents can be considered as “unclean” texts in comparison to news articles, since the writers, who are often non-professional writers, tend to use the spoken language rather than the written language. Therefore, these systems sometimes do not work like as well as they do on news articles. For example, in weblogs (or other consumer generated media), we can observe dialect (e.g. Kansai region, Japanese), variants (e.g. “$\overset{\textit{okkii}}{\text{おっきい}}$” and “$\overset{\textit{dekai}}{\text{デカい}}$)”, where both expressions are used to express that something is “$\overset{\textit{ookii}}{\text{大きい}}$ (*big*)”.), as well as onomatopeic words (e.g. “$\overset{\textit{fuwatoro}}{\text{ふわとろ}}$”). Successful recognition of these expressions would improve not only our system but also many other web applications.

### Detecting subject expressions

So far we have set aside the task of detecting subject expressions assuming that it is a specialized form of named entity recognition, which has been actively studied for a decade.

In some domains such as the restaurant domain, common nouns are used as a subject. For example, “kokoro (*heart*)” and “mangetsu (*full moon*)” are common nouns but are used as actual restaurant names in Japan. Therefore, it is not clear how subjects can successfully be identified. This evaluation is important to developing an overall opinion extraction system.

## Determining the semantic orientation of extracted opinions

For classifying or visualizing extracted opinions, the task of determining semantic orientations is also important.

We can assign a semantic orientation to an opinion based on the semantic orientation of its evaluation. For example, we consider that opinions which include positive-oriented evaluation expressions such as “*delicious, like, good*” also have a positive orientation. The same is true of negative-oriented evaluation expressions. For acquiring semantic orientations of words, many researchers have developed several methods and obtained good results [Hatzivassiloglou and McKeown, 1997; Kamps *et al.*, 2004; Takamura *et al.*, 2005].

In addition to the above case, we consider the case where we can not determine the semantic orientation based only on evaluation expressions. For example, the evaluation expression “*high*” does not have any orientation, that is, this expression has a *neutral* orientation. However, the aspect-evaluation pair “*risk–high*” has a negative orientation, and “*performance–high*” has a positive orientation. As these examples show, we need to assign the semantic orientation to opinions based not on evaluations alone but rather on aspect-evaluation pairs. On acquiring phrase-level semantic orientations, some research has been reported in recent years [Suzuki *et al.*, 2006; Takamura *et al.*, 2006]. We would like to incorporate these techniques into our overall system.

# References

[Asahara and Matusmoto, 2003] Masayuki Asahara and Yuji Matusmoto. *IPADIC version 2.7.0 users manual.* Nara Institute of Science and Technology, 2003. (in Japanese).

[Baldwin, 1995] Breck Baldwin. *CogNIAC: A Discourse Processing Engine.* PhD thesis, Department of Computer and Information Sciences, University of Pennsylvania, 1995.

[Berger *et al.*, 1996] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[Bunescu, 2003] Razvan Bunescu. Associative anaphora resolution: a web-based approach. In *Proceedings of the EACL Workshop on The Computational Treatment of Anaphora*, pages 47–52, 2003.

[Church and Hanks, 1989] Kenneth W. Church and Patrick Hanks. Word association norms, mutual information, and lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics* (*ACL*) , pages 76–83. Association for Computational Linguistics, 1989.

[Clark, 1977] Herbert H. Clark. *Bridging. Thinking: readings in cognitive science.* Cambridge : Cambridge University Press, 1977.

[Culotta and Sorensen, 2004] Aron Culotta and Jefferey Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (*ACL*) , pages 423–429, 2004.

[Dave *et al.*, 2003] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th International World Wide Web Conference* (*WWW*), pages 519–528, 2003.

[Dunning, 1993] Ted Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[Etzioni *et al.*, 2004] Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Web-scale information extraction in knowitall. In *Proceedings of the 13th International World Wide Web Conference (WWW)*, pages 100–110, 2004.

[Fellbaum, 1998] Christiane Fellbaum, editor. *WordNet-An Electronic Lexical Database*. The MIT press, 1998.

[Grosz *et al.*, 1995] Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226, 1995.

[Halliday and Hasan, 1976] Michael A. K. Halliday and Ruqaiya Hasan. *Cohesion in English*. English Language Series,Title No.9. Longman, 1976.

[Harabagiu *et al.*, 2005] Sanda M. Harabagiu, Cosmin Adrian Bejan, and Paul Morarescu. Shallow semantics for relation extraction. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence(IJCAI)*, pages 1061–1066, 2005.

[Hatzivassiloglou and McKeown, 1997] Vasileios Hatzivassiloglou and Kathleen R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics(ACL) and the 8th Conference of the European Chapter of the Association for Computational Linguistics(EACL)*, pages 174–181, 1997.

[Hofmann, 1999] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

[Hu and Liu, 2004] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 168–177, 2004.

[Iida *et al.*, 2003] Ryu Iida, Kentaro Inui, Hiroya Takamura, and Yuji Matsumoto. Incorporating contextual cues in trainable models for coreference resolution. In *Proceedings of the EACL Workshop on the Computational Treatment of Anaphora*, pages 23–30, 2003.

[Ikehara *et al.*, 1997] Satoru Ikehara, Masahiro Miyazaki, Satoshi Shirai, Akio Yokoo, Hiromi Nakaiwa, Kentaro Ogura, Yoshifumi Ooyama, and Yoshihiko Hayashi. *Goi-Taikei — A Japanese Lexicon*. Iwanami Shoten, 1997. 5 volumes.

[Kameyama, 1986] Megumi Kameyama. A property-sharing constraint in centering. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pages 200–206, 1986.

[Kamps *et al.*, 2004] Jaap Kamps, Maarten Marx, Robert J. Mokken, and Maarten de Rijke. Using wordnet to measure semantic orientations of adjectives. In *Proceedings of international conference on language resources and evaluation (LREC)* , pages 201–208, 2004.

[Kanayama and Nasukawa, 2004] Hiroshi Kanayama and Tetsuya Nasukawa. Deeper sentiment analysis using machine translation technology. In *Proc. of the 20th International Conference on Computational Linguistics(COLING)*, pages 494–500, 2004.

[Kim and Hovy, 2004] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 1367–1373, 2004.

[Kudo and Matsumoto, 2004] Taku Kudo and Yuji Matsumoto. A boosting algorithm for classification of semi-structured text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 301–308, 2004.

[Liu *et al.*, 2005] Bing Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th International World Wide Web Conference (WWW)*, pages 342–351, 2005.

[Matsumoto *et al.*, 2005] Shotaro Matsumoto, Hiroya Takamura, and Manabu Okumura. Sentiment classification using word sub-sequences and dependency sub-trees. In *Proceedings of the 9th Pacific-Asia International Conference on Knowledge Discovery and Data Mining (PAKDD)* , pages 301–310, 2005.

[Mitkov, 1997] Ruslan Mitkov. Factors in anaphora resolution: they are not the only things that matter. a case study ba sed on two different approaches. In *Proc. of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaph ora Resolution*, pages 14–21, 1997.

[Mullen and Collier, 2004] Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 412–418, 2004.

[Murano and Sato, 2003] Seiji Murano and Satoshi Sato. Automatic extraction of subjective sentences using syntactic patterns. In *Proceedings of the ninth Annual Meeting of the Association for Natural Language Processing*, pages 67–70, 2003. (in Japanese).

[Nakaiwa and Shirai, 1996] Hiromi Nakaiwa and Satoshi Shirai. Anaphora resolution of japanese zero pronouns with deictic reference. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 812–817, 1996.

[Ng and Cardie, 2002] Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 104–111, 2002.

[Ng, 2004] Vincent Ng. Learning noun phrase anaphoricity to improve coreference resolution: Issues in representation and optimization. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (*ACL*) , pages 152–159, 2004.

[Pang and Lee, 2004] Bo Pang and Lillian Lee. A sentiment education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (*ACL*) , pages 271–278, 2004.

[Pang *et al.*, 2002] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In

*Proceedings of the Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 79–86, 2002.

[Poesio *et al.*, 2004] Massimo Poesio, Rahul Mehta, Axel Maroudas, and Janet Hitzeman. Learning to resolve bridging references. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 143–150, 2004.

[Popescu and Etzioni, 2005] Ana-Maria Popescu and Oren Etzioni. Extracting product features and opinions from reviews. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language* (*HLT/EMNLP*), pages 339–346, 2005.

[Riloff and Jones, 1999] Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on Artificial Intelligence* (*AAAI*), pages 474–479, 1999.

[Riloff *et al.*, 2003] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the 7th Conference on Natural Language Learning* (*CoNLL*), pages 25–32, 2003.

[Sebastiani, 2002] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[Soon *et al.*, 2001] Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, 2001.

[Stone *et al.*, 1966] Philip J. Stone, Dexter C. Dunphy, and Daniel M. Ogilvie Marshall S. Smith. *The General Inquirer: A Computer Approach to Content Analysis.* The MIT Press, 1966.

[Suzuki *et al.*, 2006] Yasuhiro Suzuki, Hiroya Takamura, and Manabu Okumura. Application of semi-supervised learning to evaluative expression classification. In *Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics* (*CICLing*), pages 502–513, 2006.

[Takamura *et al.*, 2005] Hiroya Takamura, Takashi Inui, and Manabu Okumura. Extracting semantic orientations of words using spin model. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (*ACL*) , pages 133–140, 2005.

[Takamura *et al.*, 2006] Hiroya Takamura, Takashi Inui, and Manabu Okumura. Latent variable models for semantic orientations of phrases. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics* (*EACL*) , pages 201–208, 2006.

[Tateishi *et al.*, 2001] Kenji Tateishi, Yoshihide Ishiguro, and Toshikazu Fukushima. Opinion information retrieval from the internet. In *IPSJ SIGNL Note 144-11*, pages 75–82, 2001. (in Japanese).

[Tateishi *et al.*, 2004] Kenji Tateishi, Toshikazu Fukushima, Nozomi Kobayashi, Tetsuro Takahashi, Atsushi Fujita, Kentaro Inui, and Yuji Matsumoto. Web opinion extraction and summarization based on viewpoints of products. In *IPSJ SIGNL Note 163*, pages 1–8, 2004. (in Japanese).

[Thelen and Riloff, 2002] Michael Thelen and Ellen Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 214–221, 2002.

[Turney and Littman, 2002] Peter D. Turney and M. L. Littman. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. Technical report, National Research Council, Institute for Information Technology, ERB-1094, 2002.

[Turney, 2002] Peter D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pages 417–424, 2002.

[Vapnik, 1998] V. N. Vapnik. *Statistical Learning Theory.* Adaptive and Learning Systems for Signal Processing Communicat ions, and control. John Wiley & Sons, 1998.

[Wiebe *et al.*, 2005] Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39:165–210, 2005.

[Wiebe, 2002] Janyce Wiebe. Instructions for annotating opinions in newspaper articles. Technical report, Department of comuputer science, University of Pittsburgh, TR-02-101, 2002.

[Wilson *et al.*, 2005] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language* (*HLT/EMNLP*), pages 347–354, 2005.

[Yi *et al.*, 2003] Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proceedings of the third IEEE International Conference on Data Mining* (*ICDM*), pages 427–434, 2003.

[Yu and Hatzivassiloglou, 2003] Hong Yu and Vasileios Hatzivassiloglou. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pages 129–136, 2003.

[Zelenko *et al.*, 2003] Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel method for relation extraction. In *Journal of Machine Learning research*, pages 1083–106, 2003.

# List of Publication

## Journal Papers

1. Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. Opinion Mining from Web documents: Extraction and Structurization. Journal of Japanese society for artificial intelligence, Vol.22 No.2, special issue on data mining and statistical science, pages 227–238, 2007.

2. Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Kenji Tateishi, and Toshikazu Fukushima. Collecting evaluative expressions for opinion extraction. Journal of the Association for Natural Language Processing of Japan, Vol.12, No.2, pages 203–222, 2005. (in Japanese).

## International Conference and Workshops

1. Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto Opinion Mining on the Web by Extracting Subject-Attribute-Value Relations. In *Proceedings of AAAI Spring Symposia on Computational Approaches to Analyzing Weblogs (AAAI-CAAW)*, pages 86–91, 2006.

2. Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Opinion Extraction Using a Learning-Based Anaphora Resolution Technique. In *the Second International Joint Conference on Natural Language Processing (IJCNLP), Companion Volume to the Proceeding of Conference including Posters/Demos and Tutorial Abstracts*, pages 175–180, 2005.

3. Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Opinion

Mining as Extraction of Attribute-Value Pairs. In *Proceedings of the First International Workshop on Risk Management Systems with Intelligent Data Analysis (RDMDA-2005)*, pages 89–98, 2005.

4. Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Kenji Tateishi, and Toshikazu Fukushima. Collecting Evaluative Expressions for Opinion Extraction In *Proceedings of the First International Joint Conference on Natural Language Processing (IJCNLP-04)*, pages 584–589, 2004.

# Other Publications

1. Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. Opinion mining from weblogs: extraction and structurization. In *Proceedings of the International Workshop on Data-Mining and Statistical Science (DMSS2006)*, pages 85–92, 2006.

2. Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Opinion Mining as Extraction of Attribute-Value Relations. Lecture Notes in Artificial Intelligence, Vol. 4012, pages 470-481, Springer-Verlag, 2006.

3. Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. Identifying aspect-aspect/subject-aspect relations for opinion extraction. In *Proceedings of The 12th Annual Meeting of The Association for Natural Language*, 2006. (in Japanese).

4. Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. Designing the task of opinion extraction and structurization. In *IPSJ SIGNL Note 171-18*, pages 111–118, 2006. (in Japanese).

5. Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Extracting attribute-value pairs and its opinion-hood using an approach to anaphora resolution In *Proceedings of The 11th Annual Meeting of The Association for Natural Language*, pages 436-439, 2005. (in Japanese).

6. Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Kenji Tateishi, and Toshikazu Fukushima. Collecting Evaluative Expressions for Opinion Ex-

traction. Lecture Notes in Artificial Intelligence, Vol. 3248, pages 596-605, Springer-Verlag, 2005.

7. Ryu Iida, Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Kenji Tateishi, and Toshikazu Fukushima. A machine learning-based method to extract attribute-value pairs for opinion mining. In *IPSJ SIGNL Note 165-4*, pages 21–28, 2005. (in Japanese).

8. Nozomi Kobayashi, Kentaro Inui, Yuji Matsumoto, Kenji Tateishi, and Toshikazu Fukushima. Collecting evaluative expressions by a text mining technique. In *IPSJ SIGNL Note 154-12*, pages 77–84, 2003. (in Japanese).

9. Nozomi Kobayashi, Takashi Inui, and Kentaro Inui. Dictionary-based acquisition of the lexical knowledge for p/n analysis. In *Proceedings of Japanese Society for Artificial Intelligence, SLUD-33*, pages 45–50, 2001. (in Japanese).

# Acknowledgements

(英語では感謝の気持ちを表すことが難しいので日本語で)

はじめに，本研究を行う機会をくださった指導教官の松本裕治教授に感謝いたします．多忙な中でも時間を割いていただき，研究を進める上での的確なコメントを数多くいただきました．心から感謝いたします．

乾健太郎助教授には，学部のときから 6 年間，研究に対するコメントのみならず，研究者としての心構えについて学ばせていただきました．本当にありがとうございました．

また，審査委員として本論文を査読していただき，有益なコメントをいただきました植村俊亮教授，鹿野清宏教授に心よりお礼申しあげます．また，2006 年をもって退官された渡邉勝正先生には，修士を含め博士中間報告までに多くのコメントをいただきました．ここに心よりお礼申しあげます．

筑波大学の Edson T. Miyamoto 講師，新保仁助手，浅原正幸助手には，研究会や勉強会を通していろいろとお世話になりました．ありがとうございました．

本論文の 4 章の実験および 5 章の実験データは，共同研究として NEC インターネットシステム研究所の福島俊一氏，立石健二氏に協力していただきました．ここに感謝いたします．

本研究に必要なデータの注釈付け作業を北川祐子さん，中川由樹子さんにお願いいたしました．ありがとうございました．また，秘書の大川基子さんには，研究活動を進める上でいろいろとフォローしていただきました．本当にありがとうございました．

奈良先端科学技術大学院大学 自然言語処理学講座の皆様，九州工業大学 元乾研究室の皆様には先輩，同輩，後輩含めていろいろなことを学ばせていただきました．ここに感謝の意をあらわします．

特に，九州工業大学在学時から，研究に関して相談にのっていただいた乾孝司さん，高橋哲朗さん，藤田篤さんに心よりお礼申しあげます．

同期の伊藤敬彦さん，飯田龍さん，徳永泰浩さん，山崎貴宏さんにはいろいろとお世話になりました．ここまでこれたのは同期の皆がいてくれたからです．本当にありがとうございました．

歴代の計算機係さん，特に小町守さんにはさまざまな問題の解決をしていただきました．ありがとうございました．

Campbell Hore さんと Eric Nichols さんには英語の校正で大変お世話になりま

した．この論文もお二人のチェックあってのものです．ありがとうございました．

最後に，博士課程までの進学を許し，温かく見守ってくれた両親，さやか姉と求に心からの感謝を．