# Deeper Sentiment Analysis Using Machine Translation Technology

**KANAYAMA Hiroshi** **NASUKAWA Tetsuya**
**WATANABE Hideo**
Tokyo Research Laboratory, IBM Japan, Ltd.
1623-14 Shimotsuruma, Yamato-shi, Kanagawa-ken, 242-8502 Japan
{hkana,nasukawa,hiwat}@jp.ibm.com

## Abstract

This paper proposes a new paradigm for sentiment analysis: translation from text documents to a set of sentiment units. The techniques of deep language analysis for machine translation are applicable also to this kind of text mining task. We developed a high-precision sentiment analysis system at a low development cost, by making use of an existing transfer-based machine translation engine.

## 1 Introduction

*Sentiment analysis* (SA) (Nasukawa and Yi, 2003; Yi et al., 2003) is a task to obtain writers' feelings as expressed in positive or negative comments, questions, and requests, by analyzing large numbers of documents. SA is becoming a useful tool for the commercial activities of both companies and individual consumers, because they want to sort out opinions about products, services, or brands that are scattered in online texts such as product review articles, replies given to questionnaires, and messages in bulletin boards on the WWW.

This paper describes a method to extract a set of *sentiment units* from sentences, which is the key component of SA. A sentiment unit is a tuple of a sentiment[1], a predicate, and its arguments. For example, these sentences in a customer's review of a digital camera (1) contained three sentiment units (1a), (1b), and (1c). Apparently these units indicate that the camera has good features in its lens and recharger, and a bad feature in its price.

It has excellent lens, but the price is too high.
I don't think the quality of the recharger
has any problem. (1)

[favorable] excellent (lens) (1a)
[unfavorable] high (price) (1b)
[favorable] problematic$^{+\text{neg}}$ (recharger) (1c)

The extraction of these sentiment units is not a trivial task because many syntactic and semantic operations are required. First, the structure of a predicate and its arguments may be changed from the syntactic form as in (1a) and (1c). Also modal, aspectual, and negation information must be handled, as in (1c). Second, a sentiment unit should be constructed as the smallest possible informative unit so that it is easy to handle for the organizing processes after extraction. In (1b) the degree adverb 'too' is omitted to normalize the expression. For (1c), the predicate 'problematic' has the argument 'recharger' instead of the head word of the noun phrase 'the quality of the recharger', because just using 'quality' is not informative to describe the sentiment of the attribute of a real-world object. Moreover, disambiguation of sentiments is necessary: in (1b) the adjective 'high' has the 'unfavorable' feature, but 'high' can be treated as 'favorable' in the expression "resolution is high".

We regard this task as *translation from text to sentiment units*, because we noticed that the deep language analysis techniques which are required for the extraction of sentiment units are analogous to those which have been studied for the purpose of language translation. We implemented an accurate sentiment analyzer by making use of an existing transfer-based machine translation engine (Watanabe, 1992), replacing the translation patterns and bilingual lexicons with sentiment patterns and a sentiment polarity lexicon. Although we used many techniques for deep language analysis, the system was implemented at a surprisingly low development cost because the techniques for machine translation could be reused in the architecture described in this paper.

We aimed at the high precision extraction of sentiment units. In other words, our SA system attaches importance to each individual sentiment expression, rather than to the quantitative tendencies of reputation. This is in order to meet the requirement of the SA users who want to know not only the overall goodness of an object, but also the breakdown of opinions. For example, when there are many positive opinions and only one negative opinion, the negative one should not be ignored because of its low percentage, but should be investigated thoroughly since valuable knowledge is often found in such a minority opinion. Figure 1 illustrates an image of the SA output. The outliner organizes positive and negative opinions by topic words, and provides references to the original text.

[1] Possible values of a sentiment are 'favorable', 'unfavorable', 'question', and 'request'. In this paper the discussion is mostly focused on the first two values.

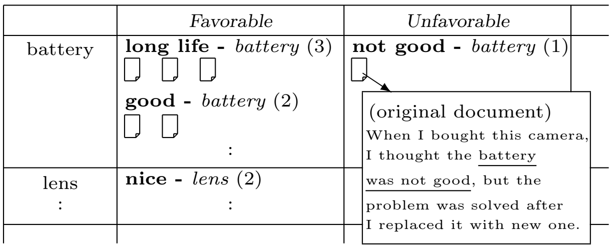

Figure 1: An image of an outliner which uses SA output. Users can refer to the original text by clicking on the document icons.

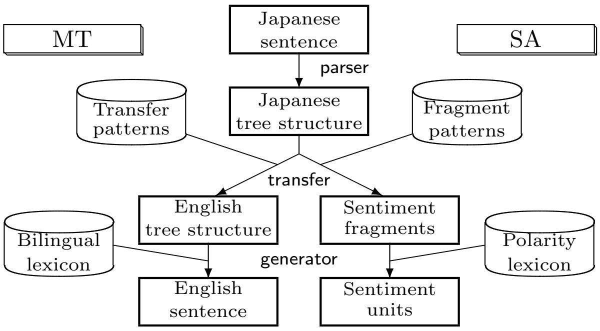

Figure 2: The concept of the machine translation engine and the sentiment analyzer. Some components are shared between them. Also other components are similar between MT and SA.

This means that the approach for SA should be switched from the rather shallow analysis techniques used for text mining (Hearst, 1999; Nasukawa and Nagano, 2001), where some errors can be treated as noise, into deep analysis techniques such as those used for machine translation (MT) where all of the syntactic and semantic phenomena must be handled.

We implemented a Japanese SA system using a Japanese to English translation engine. Figure 2 illustrates our SA system, which utilizes a MT engine, where techniques for parsing and pattern matching on the tree structures are shared between MT and SA.

Section 2 reviews previous studies of sentiment analysis. In Section 3 we define the sentiment unit to be extracted for sentiment analysis. Section 4 presents the implementation of our system, comparing the operations and resources with those used for machine translation. Our system is evaluated in Section 5. In the rest of paper we mainly use Japanese examples because some of the operations depend on the Japanese language, but we also use English examples to express the sentiment units and some language-independent issues, for understandability.

## 2 Previous work on Sentiment Analysis

Some prior studies on sentiment analysis focused on the document-level classification of sentiment (Turney, 2002; Pang et al., 2002) where a document is assumed to have only a single sentiment, thus these studies are not applicable to our goal. Other work (Subasic and Huettner, 2001; Morinaga et al., 2002) assigned sentiment to words, but they relied on quantitative information such as the frequencies of word associations or statistical predictions of favorability.

Automatic acquisition of sentiment expressions have also been studied (Hatzivassiloglou and McKeown, 1997), but limited to adjectives, and only one sentiment could be assigned to each word.

Yi et al. (2003) pointed out that the multiple sentiment aspects in a document should be extracted. This paper follows that approach, but exploits deeper analysis in order to avoid the analytic failures reported by Nasukawa and Yi (2003), which occurred when they used a shallow parser and only addressed a limited number of syntactic phenomena. In our in-depth approach described in the next section, two types of errors out of the four reported by Nasukawa and Yi (2003) were easily removed[2].

## 3 Sentiment Unit

This section describes the sentiment units which are extracted from text, and their roles in the sentiment analysis and its applications.

A sentiment unit consists of a sentiment, a predicate, its one or more arguments, and a surface form. Formally it is expressed as in Figure 3.

The 'sentiment' feature categorizes a sentiment unit into four types: 'favorable' [fav], 'unfavorable' [unf], 'question' [qst], and 'request' [req]. A predicate is a word, typically a verb or an adjective, which conveys the main notion of the sentiment unit. An argument is also a word, typically a noun, which modifies the predicate with a case postpositional in Japanese. They roughly correspond to a subject and an object of the predicate in English.

For example, from the sentence (2)[3], the extracted sentiment unit is (2a).

| | | | |
|---|---|---|---|
| *ABC123-ha* | *renzu-ga* | *subarashii.* | |
| ABC123-TOPIC | lens-NOM | excellent | (2) |
| 'ABC123 has an excellent lens.' | | | |

[fav] excellent ⟨ ABC123, lens ⟩ (2a)

The sentiment unit (2a) stands for the sentiment is 'favorable', the predicate is 'excellent' and its arguments are 'ABC123' and 'lens'. In this case, both 'ABC123' and 'lens' are counted as words which are associated with a favorable sentiment. Arguments are used as the keywords in the outliner, as in the leftmost column in Figure 1. Predicates with no argument are ignored, because they have no effects on the view and often become noise.

[2] Though this paper handles Japanese SA, we also implemented an English version of SA using English-French translation techniques, and that system solved the problems which were mentioned in Nasukawa and Yi's paper.

[3] 'ABC123' is a fictitious product name.

```
<sentiment unit> ::= <sentiment> <predicate> <argument>+ <surface>
<sentiment> ::= favorable | unfavorable | question | request
<predicate> ::= <word> <feature>*
<argument> ::= <word> <feature>*
<surface> ::= <string>
```

Figure 3: The definition of a sentiment unit.

The predicate and its arguments can be different from the surface form in the original text. Semantically similar representations should be aggregated to organize extracted sentiments, so the examples in this paper use English canonical forms to represent predicates and arguments, while the actual implementation uses Japanese expressions.

Predicates may have features, such as negation, facility, difficulty, etc. For example, "ABC123 doesn't have an excellent lens." brings a sentiment unit "[unf] excellent$^{+\text{neg}}$ ⟨ ABC123, lens ⟩". Also the facility/difficulty feature affects the sentiments such as "[unf] break$^{+\text{facil}}$" for 'easy to break' and "[unf] learn$^{+\text{diff}}$" 'difficult to learn'.

The surface string is the corresponding part in the original text. It is used for reference in the view of the output of SA, because the surface string is the most understandable notation of each sentiment unit for humans.

We use the term *sentiment polarity* for the selection of the two sentiments [fav] and [unf]. The other two sentiments, [qst] and [req] are important in applications, *e.g.* the automatic creation of FAQ. Roughly speaking, [qst] is extracted from an interrogative sentence, and [req] is used for imperative sentences or expressions such as "I want ..." and "I'd like you to ...". From a pragmatic point of view it is difficult to distinguish between them[4], but we classify them using simple rules.

## 4 Implementation

This section describes operations and resources designed for the extraction of sentiment units. There are many techniques analogous to those for machine translation, so first we show the architecture of the transfer-based machine translation engine which is used as the basis of the extraction of sentiment units.

### 4.1 Transfer-based Machine Translation Engine

As illustrated on the left side of Figure 2, the transfer-based machine translation system consists of three parts: a source language syntactic parser, a bilingual transfer which handles the syntactic tree structures, and a target language generator. Here the flow of the Japanese to English translation is shown with the following example sentence (3).

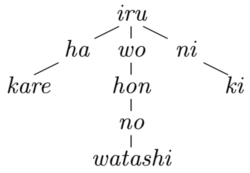

Figure 4: The Japanese syntactic tree for the sentence (3).

*Kare-ha* *watashi-no*
He-TOPIC I-GEN
*hon-wo* *ki-ni* *iru.*
book-ACC mind-DAT enter (3)
'He likes my book.'

First the syntactic parser parses the sentence (3) to create the tree structure as shown in Figure 4.

Next, the transfer converts this Japanese parse tree into an English one by applying the translation patterns as in Figure 5. A translation pattern consists of a tree of the source language, a tree of the target language, and the word correspondences between both languages.

The patterns (a) and (b) in Figure 5 match with the subtrees in Figure 4, as Figure 6 illustrates. This matching operation is very complicated because there can be an enormous number of possible combinations of patterns. The fitness of the pattern combinations is calculated according to the similarity of the source tree and the left side of the translation pattern, the specificity of the translation pattern, and so on. This example also shows the process of matching the Japanese case markers (postpositional particles). The source tree and the pattern (a) match even though the postpositional particles are different ('*ha*' and '*ga*'). This process may be much more complicated when a verb is transformed into special forms *e.g.* passive or causative. Besides this there are many operations to handle syntactic and semantic phenomena, but here we take them for granted because of space constraints.

Now the target fragments have been created as in Figure 6, using the right side of the matched translation patterns as in Figure 5. The two fragments are attached at the shared node '$\boxed{\text{noun}_2}$', and lexicalized by using the bilingual lexicon. Finally the target sentence "He likes my book." is generated by the target language generator.

[4] For example, the interrogative sentence "Would you read it?" implies a request.

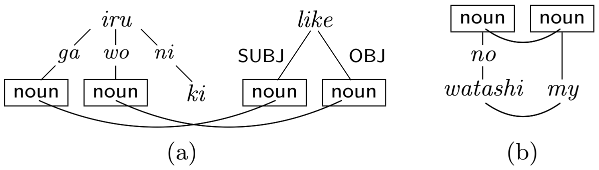

Figure 5: Two examples of Japanese-English translation patterns. The left side and the right side are Japanese and English syntactic trees, respectively. The '$\boxed{\text{noun}}$' works as a wildcard which matches with any noun. Curves stand for correspondences between Japanese and English words.

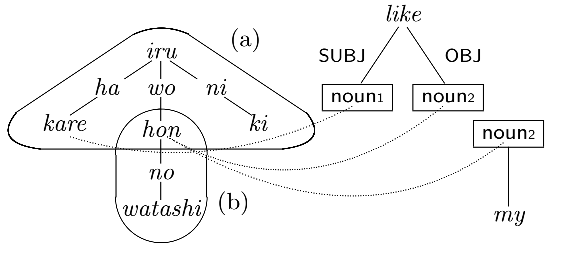

Figure 6: Transferring the Japanese tree in Figure 4 into the English tree. The patterns in Figure 5 create two English fragments, and they are attached at the nodes '$\boxed{\text{noun}_2}$' which share the same correspondent node in the source language tree.

### 4.2 Techniques Required for Sentiment Analysis

Our aim is to extract sentiment units with high precision. Moreover, the set of arguments of each predicate should be selected necessarily and sufficiently. Here we show that the techniques to meet these requirements are analogous to the techniques for machine translation which have been reviewed in Section 4.1.

#### 4.2.1 Full parsing and top-down tree matching

Full syntactic parsing plays an important role to extract sentiments correctly, because the local structures obtained by a shallow parser are not always reliable. For example, expressions such as "I don't think *X* is good", "I hope that *X* is good" are not favorable opinions about *X*, even though "*X* is good" appears on the surface. Therefore we use top-down pattern matching on the tree structures from the full parsing in order to find each sentiment fragment, that is potentially a part of a sentiment unit.

In our method, initially the top node is examined to see whether or not the node and its combination of children nodes match with one of the patterns in the pattern repository. In this top-down manner, the nodes "don't think" and "hope" in the above examples are examined before "*X* is good", and thus the above expressions won't be misunderstood to express favorable sentiments.

There are three types of patterns: principal patterns, auxiliary patterns, and nominal patterns. Figure 7 illustrates examples of principal patterns: the pattern (c) converts a Japanese expression "$\boxed{\text{noun}}$-*ga warui*" to a sentiment unit "[unf] bad ⟨ $\boxed{\text{noun}}$ ⟩". The pattern (d) converts an expression "$\boxed{\text{noun}}$-*wo ki-ni iru*" to a sentiment unit "[fav] like ⟨ $\boxed{\text{noun}}$ ⟩", where the subject (the noun preceding the postpositional *ga*) is excluded from the arguments because the subject of 'like' is usually the author, who is not the target of sentiment analysis.

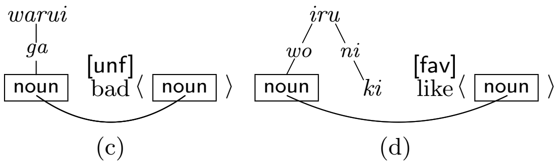

Figure 7: Examples of principal patterns.

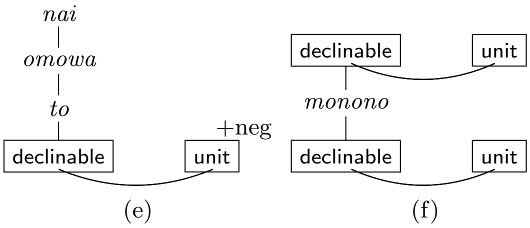

Figure 8: Examples of auxiliary patterns. '$\boxed{\text{declinable}}$' denotes a verb or an adjective in Japanese. Note that the two $\boxed{\text{unit}}$s on the right side of (f) are not connected. This means two separated sentiment units can be obtained.

Another type is the auxiliary pattern, which expands the scope of matching. Figure 8 has two examples. The pattern (e) matches with phrases such as "*X-wa yoi-to omowa-nai.* ((I) don't think *X* is good.)" and produces a sentiment unit with the negation feature. When this pattern is attached to a principal pattern, its favorability is inverted. The pattern (f) allows us to obtain two separate sentiment units from sentences such as "*Dezain-ga warui-mononono, sousasei-ha yoi.* (The design is bad, but the usability is good.)".

#### 4.2.2 Informative noun phrase

The third type of pattern is a nominal pattern. Figure 9 shows three examples. The pattern (g) is used to avoid a formal noun (nominalizer) being an argument. Using this pattern, from the sentence "*Kawaii no-ga suki-da.* ((I) like pretty things)", "[fav] like ⟨ pretty ⟩" can be extracted instead of "[fav] like ⟨ thing ⟩". The pattern (h) is used to convert a noun phrase "*renzu-no shitsu* (quality of the lens)" into just "lens". Due to this operation, from Sentence (4), an informative sentiment unit (4a) can be obtained instead of a less informative one (4b).

*Renzu-no* *shitsu-ga* *yoi.*
lens-GEN quality-NOM good (4)
'The quality of the lens is good.'

[fav] good ⟨ lens ⟩ (4a)
? [fav] good ⟨ quality ⟩ (4b)

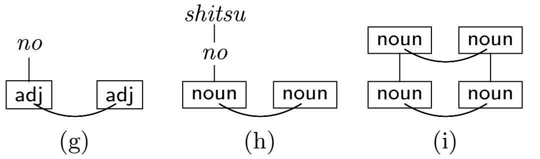

Figure 9: Examples of nominal patterns.

The pattern (i) is for compound nouns such as "*juuden jikan* (recharging time)". A sentiment unit "long ⟨ time ⟩" is not informative, but "long ⟨ recharging time ⟩" can be regarded as a [unf] sentiment.

### 4.2.3 Disambiguation of sentiment polarity

Some adjectives and verbs may be used for both favorable and unfavorable predicates. This variation of sentiment polarity can be disambiguated naturally in the same manner as the word sense disambiguation in machine translation. The adjective '*takai* (high)' is a typical example, as in (5a) and (5b). In this case the sentiment polarity depends on the noun preceding the postpositional particle '*ga*': favorable if the noun is '*kaizoudo* (resolution)', unfavorable if the noun is a product name. The semantic category assigned to a noun holds the information used for this type of disambiguation.

*Kaizoudo-ga* *takai.*
resolution-NOM high → [fav] (5a)
'The resolution is high.'

*ABC123-ga* *takai.*
ABC123-NOM high (price) → [unf] (5b)
'ABC123 is expensive.'

### 4.2.4 Aggregation of synonymous expressions

In contrast to disambiguation, aggregation of synonymous expressions is important to organize extracted sentiment units. If the different expressions which convey the same (or similar) meanings are aggregated into a canonical one, the frequency increases and one can easily find frequently mentioned opinions.

Using the translation architecture, any forms can be chosen as the predicates and arguments by adjusting the patterns and lexicons. That is, monolingual word translation is done in our method.

### 4.3 Resources for Sentiment Analysis

We prepared the following resources for sentiment analysis:

**Principal patterns:** The verbal and adjectival patterns for machine translation were converted to principal patterns for sentiment analysis. The left sides of the patterns are compatible with the source language parts of the original patterns, so we just assigned a sentiment polarity to each word. A total of 3752 principal patterns were created.

**Auxiliary/Nominal patterns:** A total of 95 auxiliary patterns and 36 nominal patterns were created manually.

**Polarity lexicon:** Some nouns were assigned sentiment polarity, *e.g.* [unf] for 'noise'. This polarity is used in expressions such as "*... ga ooi.* (There are many ...)". This lexicon is also used for the aggregation of words.

Some patterns and lexicons are domain-dependent. The situation is the same as in machine translation. Fortunately the translation engine used here has a function to selectively use domain-dependent dictionaries, and thus we can prepare patterns which are especially suited for the messages on bulletin boards, or for the domain of digital cameras. For example, "The size is small." is a desirable feature of a digital camera. We can assign the appropriate sentiment (in this case, [fav]) by using a domain-specific principal pattern.

## 5 Evaluation

We conducted two experiments on the extraction of sentiment units from bulletin boards on the WWW that are discussing digital cameras. A total of 200 randomly selected sentences were analyzed by our system. The resources were created by looking at other parts of the same domain texts, and therefore this experiment is an open test.

Experiment 1 measured the precision of the sentiment polarity, and Experiment 2 evaluated the informativeness of the sentiment units. In this section we handled only the sentiments [fav] and [unf] sentiments, thus the other two sentiments [qst] and [req] were not evaluated.

### 5.1 Experiment 1: Precision and Recall

In order to see the reliability of the extracted sentiment polarities, we evaluated the following three metrics:

**Weak precision:** The coincidence rate of the sentiment polarity between the system's output and manual output when both the system and the human evaluators assigned either a favorable or unfavorable sentiment.

**Strong precision:** The coincidence rate of the sentiment polarity between the system's output and manual output when the system assigned either a favorable or unfavorable sentiment.

**Recall:** The detection rate of sentiment units within the manual output.

These metrics are measured by using two methods: (A) our proposed method based on the machine translation engine, and (B) the lexicon-only method, which emulates the shallow parsing approach. The latter method used the simple polarity lexicon of adjectives and verbs, where an adjective or a verb had only one sentiment polarity, then no disambiguation was done. Except for the direct negation of

|  | (A) MT |  | (B) Lexicon only |  |
|---|---|---|---|---|
| Weak prec. | 100% | (31/31) | 80% | (41/51) |
| Strong prec. | 89% | (31/35) | 44% | (41/93) |
| Recall | 43% | (31/72) | 57% | (41/72) |

Table 1: Precision and recall for the extraction of sentiment units from 200 sentences.

(A) MT

|  |  | Manual |  |  |
|---|---|---|---|---|
|  |  | f | n | u |
| System | f | **20** | 3 | 0 |
|  | n | 27 | - | 14 |
|  | u | 0 | 1 | **11** |

(B) Lexicon only

|  |  | Manual |  |  |
|---|---|---|---|---|
|  |  | f | n | u |
| System | f | **26** | 19 | 6 |
|  | n | 14 | - | 7 |
|  | u | 4 | 23 | **15** |

Table 2: The breakdown of the results of Experiment 1. The columns and rows show the manual output and the system output, respectively (f: favorable, n: non-sentiment, u: unfavorable). The sum of the bold numbers equals the numerators of the precision and recall.

an adjective or a verb[5], no translation patterns were used. Instead of the top-down pattern matching, sentiment units were extracted from any part of the tree structures (the results of full-parsing were used also here).

Table 1 shows the results. With the MT framework, the weak precision was perfect, and also the strong precision was much higher, while the recall was lower than for the lexicon-only method. Their breakdowns in the two parts of Table 2 indicate that most of errors where the system wrongly assigned either of sentiments (*i.e.* human regarded an expression as non-sentiment) have been reduced with the MT framework.

All of the above results are consistent with intuition. The MT method outputs a sentiment unit only when the expression is reachable from the root node of the syntactic tree through the combination of sentiment fragments, while the lexicon-only method picks up sentiment units from any node in the syntactic tree. The sentence (6) is an example where the lexicon-only method output the wrong sentiment unit (6a). The MT method did not output this sentiment unit, thus the precision values of the MT method did not suffer from this example.

... *gashitsu-ga kirei-da-to iu hyouka-ha*
*uke-masen-deshi-ta.* (6)
'There was no opinion that the picture was sharp.'

* [fav] clear ⟨ picture ⟩ (6a)

In the lexicon-only method, some errors occurred due to the ambiguity in sentiment polarity of an adjective or a verb, *e.g.* "*Kanousei-ga takai.* (Capabilities are high.)" since '*takai* (high/expensive)' is always assigned the [unf] feature.

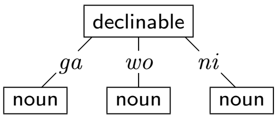

Figure 10: A naïve predicate-argument structure used by the system (C). Nouns preceding three major postpositional particles '*ga*', '*wo*', and '*ni*' are supported as the slots of arguments. On the other hand, in the system (A), there are over 3,000 principal patterns that have information on appropriate combinations for each verb and adjective.

|  | (A) MT | (C) Naïve |
|---|---|---|
| Less redundant | 2/35 | 0/35 |
| More informative | 13/35 | 1/35 |
| Both | 1/35 | 0/35 |

Table 3: Comparison of scope of sentiment units. The numbers mean the counts of the better output for each system among 35 sentiment units. The remainder is the outputs that were the same in both systems.

The recall was not so high, especially in the MT method, but according to our error analysis the recall can be increased by adding auxiliary patterns. On the other hand, it is almost impossible to increase the precision without our deep analysis techniques. Consequently, our proposed method outperforms the shallow (lexicon-only) approach.

### 5.2 Experiment 2: Scope of Sentiment Unit

We also compared the appropriateness of the scope of the extracted sentiment units between (A) the proposed method with the MT framework and (C) a method that supports only naïve predicate-argument structures as in Figure 10 and doesn't use any nominal patterns.

According to the results shown in Table 3, the MT method produced less redundant or more informative sentiment units than did relying on the naïve predicate-argument structures in about half of the cases among the 35 extracted sentiment units.

The following example (7) is a case where the sentiment unit output by the MT method (7a) was less redundant than that output by the naïve method (7b). The translation engine understood that the phrase '*kyonen-no 5-gatsu-ni* (last May)' held temporal information, therefore it was excluded from the arguments of the predicate 'enhance', while both 'function' and 'May' were the arguments of 'enhance' in (7b). Apparently the argument 'May' is not necessary here.

... *kyonen-no 5-gatsu-ni kinou-ga*
*kairyou-sare-ta you-desu.* (7)
'It seems the function was enhanced last May.'

[fav] enhance ⟨ function ⟩ (7a)
? [fav] enhance ⟨ function, May ⟩ (7b)

Example (8) is another case where the sentiment

[5] "He doesn't like it." is regarded as negation, but "I don't think it is good." is *not*.

unit output by the MT method (8a) was more informative than that output by the naïve method (8b). Than the Japanese functional noun '*hou*', its modifier 'zoom' was more informative. The MT method successfully selected the noun 'zoom' as the argument of 'desirable'.

... *zuum-no hou-ga nozomashii.* (8)
'A zoom is more desirable.'

| | | |
|---|---|---|
| | [fav] desirable ⟨ zoom ⟩ | (8a) |
| ? | [fav] desirable ⟨ *hou* ⟩ | (8b) |

The only one case we encountered where the MT method extracted a less informative sentiment unit was the sentence "*Botan-ga satsuei-ni pittari-desu* (The shutter is suitable for taking photos)". The naïve method could produce the sentiment unit "[fav] suitable ⟨ shutter, photo ⟩", but the MT method created "[fav] suitable ⟨ shutter ⟩". This is due to the lack of a noun phrase preceding the post-positional particle '*ni*' in the principal pattern. Such problems can be avoided by modifying the patterns, and thus the effect of the combination of patterns for SA has been shown here.

## 6 Conclusion

This paper has proposed a new approach to sentiment analysis: the translation from text to a set of semantic fragments. We have shown that the deep syntactic and semantic analysis makes possible the reliable extraction of sentiment units, and the outlining of sentiments became useful because of the aggregation of the variations in expressions, and the informative outputs of the arguments. The experimental results have shown that the precision of the sentiment polarity was much higher than for the conventional methods, and the sentiment units created by our system were less redundant and more informative than when using naïve predicate-argument structures. Even though we exploited many advantages of deep analysis, we could create a sentiment analysis system at a very low development cost, because many of the techniques for machine translation can be reused naturally when we regard the extraction of sentiment units as a kind of translation.

Many techniques which have been studied for the purpose of machine translation, such as word sense disambiguation (Dagan and Itai, 1994; Yarowsky, 1995), anaphora resolution (Mitamura et al., 2002), and automatic pattern extraction from corpora (Watanabe et al., 2003), can accelerate the further enhancement of sentiment analysis, or other NLP tasks. Therefore this work is the first step towards the integration of shallow and wide NLP, with deep NLP.

## References

Ido Dagan and Alon Itai. 1994. Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics*, 20(4):563–596.

Vasileios Hatzivassiloglou and Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL*, pages 174–181.

Marti A. Hearst. 1999. Untangling text data mining. In *Proc. of the 37th Annual Meeting of the Association for Computational Linguistics.*

Teruko Mitamura, Eric Nyberg, Enrique Torrejon, Dave Svoboda, Annelen Brunner, and Kathryn Baker. 2002. Pronominal anaphora resolution in the kantoo multilingual machine translation system. In *Proc. of TMI 2002*, pages 115–124.

Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. 2002. Mining product reputations on the web. In *Proc. of the 8th ACM SIGKDD Conference.*

Tetsuya Nasukawa and Tohru Nagano. 2001. Text analysis and knowledge mining system. *IBM Systems Journal*, 40(4):967–984.

Tetsuya Nasukawa and Jeonghee Yi. 2003. Sentiment analysis: Capturing favorability using natural language processing. In *Proc. of the Second International Conferences on Knowledge Capture*, pages 70–77.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86.

Pero Subasic and Alison Huettner. 2001. Affect analysis of text using fuzzy semantic typing. *IEEE Trans. on Fussy Systems.*

Peter D. Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proc. of the 40th ACL Conf.*, pages 417–424.

Hideo Watanabe, Sadao Kurohashi, and Eiji Aramaki. 2003. Finding translation patterns from dependency structures. In Michael Carl and Andy Way, editors, *Recent Advances in Example-based Machine Translation*, pages 397–420. Kluwer Academic Publishers.

Hideo Watanabe. 1992. A similarity-driven transfer system. In *Proc. of the 14th COLING, Vol. 2*, pages 770–776.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics*, pages 189–196.

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