

# Dependency Tree-based Sentiment Classification using CRFs with Hidden Variables

Tetsuji Nakagawa\*, Kentaro Inui\*<sup>†</sup> and Sadao Kurohashi\*<sup>‡</sup>

\*National Institute of Information and Communications Technology

<sup>†</sup>Tohoku University

<sup>‡</sup>Kyoto University

tnaka@nict.go.jp, inui@ecei.tohoku.ac.jp, kuro@i.kyoto-u.ac.jp

## Abstract

In this paper, we present a dependency tree-based method for sentiment classification of Japanese and English subjective sentences using conditional random fields with hidden variables. Subjective sentences often contain words which reverse the sentiment polarities of other words. Therefore, interactions between words need to be considered in sentiment classification, which is difficult to be handled with simple bag-of-words approaches, and the syntactic dependency structures of subjective sentences are exploited in our method. In the method, the sentiment polarity of each dependency subtree in a sentence, which is not observable in training data, is represented by a hidden variable. The polarity of the whole sentence is calculated in consideration of interactions between the hidden variables. Sum-product belief propagation is used for inference. Experimental results of sentiment classification for Japanese and English subjective sentences showed that the method performs better than other methods based on bag-of-features.

## 1 Introduction

Sentiment classification is a useful technique for analyzing subjective information in a large number of texts, and many studies have been conducted (Pang and Lee, 2008). A typical approach for sentiment classification is to use supervised machine learning algorithms with bag-of-words as features (Pang et al., 2002), which is widely used in topic-based text classification. In the approach, a subjective sentence is represented as a set of words in the sentence, ignoring word order and head-modifier relation between words. However, sentiment classification is different from traditional topic-based text classification. Topic-based text classification is generally a linearly separable problem ((Chakrabarti,

2002), p.168). For example, when a document contains some domain-specific words, the document will probably belong to the domain. However, in sentiment classification, sentiment polarities can be reversed. For example, let us consider the sentence “The medicine kills cancer cells.” While the phrase *cancer cells* has negative polarity, the word *kills* reverses the polarity, and the whole sentence has positive polarity. Thus, in sentiment classification, a sentence which contains positive (or negative) polarity words does not necessarily have the same polarity as a whole, and we need to consider interactions between words instead of handling words independently.

Recently, several methods have been proposed to cope with the problem (Zaenen, 2004; Ikeda et al., 2008). However, these methods are based on flat bag-of-features representation, and do not consider syntactic structures which seem essential to infer the polarity of a whole sentence. Other methods have been proposed which utilize composition of sentences (Moilanen and Pulman, 2007; Choi and Cardie, 2008; Jia et al., 2009), but these methods use rules to handle polarity reversal, and whether polarity reversal occurs or not cannot be learned from labeled data. Statistical machine learning can learn useful information from training data and generally robust for noisy data, and using it instead of rigid rules seems useful. Wilson et al. (2005) proposed a method for sentiment classification which utilizes head-modifier relation and machine learning. However, the method is based on bag-of-features and polarity reversal occurred by content words is not handled. One issue of the approach to use sentence composition and machine learning is that only the whole sentence is labeled with its polarity in general corpora for sentiment classification, and each component of the sentence is not labeled, though such information is necessary for supervised ma-

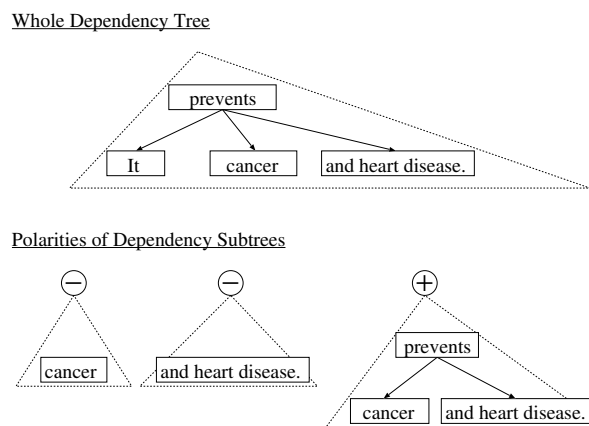


Figure 1: Polarities of Dependency Subtrees

chine learning to infer the sentence polarity from its components.

In this paper, we propose a dependency tree-based method for Japanese and English sentiment classification using conditional random fields (CRFs) with hidden variables. In the method, the sentiment polarity of each dependency subtree, which is not observable in training data, is represented by a hidden variable. The polarity of the whole sentence is calculated in consideration of interactions between the hidden variables.

The rest of this paper is organized as follows: Section 2 describes a dependency tree-based method for sentiment classification using CRFs with hidden variables, and Section 3 shows experimental results on Japanese and English corpora. Section 4 discusses related work, and Section 5 gives conclusions.

## 2 Dependency Tree-based Sentiment Classification using CRFs with Hidden Variables

In this study, we handle a task to classify the polarities (positive or negative) of given subjective sentences. In the rest of this section, we describe a probabilistic model for sentiment classification based on dependency trees, methods for inference and parameter estimation, and features we use.

### 2.1 A Probabilistic Model based on Dependency Trees

Let us consider the subjective sentence “It prevents cancer and heart disease.” In the sentence, *cancer* and *heart disease* have themselves negative polari-

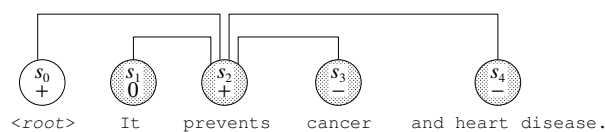


Figure 2: Probabilistic Model based on Dependency Tree

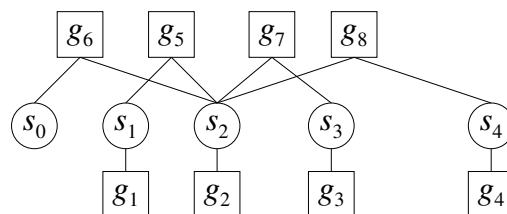


Figure 3: Factor Graph

ties. However, the polarities are reversed by modifying the word *prevents*, and the dependency subtree “prevents cancer and heart disease” has positive polarity. As a result, the whole dependency tree “It prevents cancer and heart disease.” has positive polarity (Figure 1). In such a way, we can consider the sentiment polarity for each dependency subtree of a subjective sentence. Note that we use phrases as a basic unit instead of words in this study, because phrases are useful as a meaningful unit for sentiment classification<sup>1</sup>. In this paper, a *dependency subtree* means the subtree of a dependency tree whose root node is one of the phrases in the sentence.

We use a probabilistic model as shown in Figure 2. We consider that each phrase in the subjective sentence has a random variable (indicated by a circle in Figure 2). The random variable represents the polarity of the dependency subtree whose root node is the corresponding phrase. Two random variables are dependent (indicated by an edge in Figure 2) if their corresponding phrases have head-modifier relation in the dependency tree. The node denoted as *<root>* in Figure 2 indicates a virtual phrase which represents the root node of the sentence, and we regard that the random variable of the root node is the polarity of the whole sentence. In usual annotated corpora for sentiment classification, only each sentence is labeled with its polarity, and each phrase (dependency subtree) is not labeled, so all the random variables except the one for the root node are

<sup>1</sup>From an empirical view, in our preliminary experiments with the proposed method, phrase-based processing performed better than word-based processing in accuracy and in computational efficiency.

hidden variables that cannot be observed in labeled data (indicated by gray circles in Figure 2). With such a probabilistic model, it is possible to utilize properties such that phrases which contain positive (or negative) words tend to have positive (negative) polarities, and two phrases with head-modifier relation tend to have opposite polarities if the head contains a word which reverses sentiment polarity.

Next, we define the probabilistic model as shown in Figure 2 in detail. Let  $n$  denote the number of phrases in a subjective sentence,  $w_i$  the  $i$ -th phrase, and  $h_i$  the head index of the  $i$ -th phrase. Let  $s_i$  denote the random variable which represents the polarity of the dependency subtree whose root is the  $i$ -th phrase ( $s_i \in \{+1, -1\}$ ), and let  $p$  denote the polarity of the whole sentence ( $p \in \{+1, -1\}$ ). We regard the 0-th phrase as a virtual phrase which represents the root of the sentence.  $\mathbf{w}$ ,  $\mathbf{h}$ ,  $\mathbf{s}$  respectively denote the sequence of  $w_i$ ,  $h_i$ ,  $s_i$ .

$$\begin{aligned}\mathbf{w} &= w_1 \cdots w_n, & \mathbf{h} &= h_1 \cdots h_n, & \mathbf{s} &= s_0 \cdots s_n, \\ p &= s_0.\end{aligned}$$

For the example sentence in Figure 1,  $w_1 = It$ ,  $w_2 = prevents$ ,  $w_3 = cancer$ ,  $w_4 = and\ heart\ disease.$ ,  $h_1 = 2$ ,  $h_2 = 0$ ,  $h_3 = 2$ ,  $h_4 = 2$ . We define the joint probability distribution of the sentiment polarities of dependency subtrees  $\mathbf{s}$ , given a subjective sentence  $\mathbf{w}$  and its dependency tree  $\mathbf{h}$ , using log-linear models:

$$P_{\Lambda}(\mathbf{s}|\mathbf{w}, \mathbf{h}) = \frac{1}{Z_{\Lambda}(\mathbf{w}, \mathbf{h})} \exp\left\{\sum_{k=1}^K \lambda_k F_k(\mathbf{w}, \mathbf{h}, \mathbf{s})\right\}, \quad (1)$$

$$Z_{\Lambda}(\mathbf{w}, \mathbf{h}) = \sum_{\mathbf{s}} \exp\left\{\sum_{k=1}^K \lambda_k F_k(\mathbf{w}, \mathbf{h}, \mathbf{s})\right\}, \quad (2)$$

$$F_k(\mathbf{w}, \mathbf{h}, \mathbf{s}) = \sum_{i=1}^n f_k(i, \mathbf{w}, \mathbf{h}, \mathbf{s}), \quad (3)$$

where  $\Lambda = \{\lambda_1, \dots, \lambda_K\}$  is the set of parameters of the model.  $f_k(i, \mathbf{w}, \mathbf{h}, \mathbf{s})$  is the feature function of the  $i$ -th phrase, and is classified to *node feature* which considers only the corresponding node, or *edge feature* which considers both the corresponding node and its head, as follows:

$$f_k(i, \mathbf{w}, \mathbf{h}, \mathbf{s}) = \begin{cases} f_k^n(w_i, s_i) & (k \in \mathbf{K}^n), \\ f_k^e(w_i, s_i, w_{h_i}, s_{h_i}) & (k \in \mathbf{K}^e), \end{cases} \quad (4)$$

where  $\mathbf{K}^n$  and  $\mathbf{K}^e$  respectively represent the sets of indices of node features and edge features.

## 2.2 Classification of Sentiment Polarity

Let us consider how to infer the sentiment polarity  $p \in \{+1, -1\}$ , given a subjective sentence  $\mathbf{w}$  and its dependency tree  $\mathbf{h}$ . The polarity of the root node ( $s_0$ ) is regarded as the polarity of the whole sentence, and  $p$  can be calculated as follows:

$$p = \underset{p'}{\operatorname{argmax}} P_{\Lambda}(p'|\mathbf{w}, \mathbf{h}), \quad (5)$$

$$P_{\Lambda}(p|\mathbf{w}, \mathbf{h}) = \sum_{\mathbf{s}: s_0=p} P_{\Lambda}(\mathbf{s}|\mathbf{w}, \mathbf{h}). \quad (6)$$

That is, the polarity of the subjective sentence is obtained as the marginal probability of the root node polarity, by summing the probabilities for all the possible configurations of hidden variables. However, enumerating all the possible configurations of hidden variables is computationally hard, and we use sum-product belief propagation (MacKay, 2003) for the calculation.

Belief propagation enables us to efficiently calculate marginal probabilities. In this study, the graphical model to be solved has a tree structure (identical to the syntactic dependency tree) which has no loops, and an exact solution can be obtained using belief propagation. Dependencies among random variables in Figure 2 are represented by a factor graph in Figure 3. The factor graph consists of variable nodes  $s_i$  indicated by circles, and factor (feature) nodes  $g_i$  indicated by squares. In the example in Figure 3,  $g_i (1 \leq i \leq 4)$  correspond to the node features in Equation (4), and  $g_i (5 \leq i \leq 8)$  correspond to the edge features. In belief propagation, marginal distribution is calculated by passing messages (beliefs) among the variables and factors connected by edges in the factor graph (Refer to (MacKay, 2003) for detailed description of belief propagation).

## 2.3 Parameter Estimation

Let us consider how to estimate model parameters  $\Lambda$ , given  $L$  training examples  $D = \{\langle \mathbf{w}^l, \mathbf{h}^l, p^l \rangle\}_{l=1}^L$ . In this study, we use the maximum a posteriori estimation with Gaussian priors for parameter estimation. We define the following objective function  $\mathcal{L}_{\Lambda}$ ,

and calculate the parameters  $\hat{\Lambda}$  which maximize the value:

$$\mathcal{L}_\Lambda = \sum_{l=1}^L \log P_\Lambda(p^l | \mathbf{w}^l, \mathbf{h}^l) - \frac{1}{2\sigma^2} \sum_{k=1}^K \lambda_k^2, \quad (7)$$

$$\hat{\Lambda} = \underset{\Lambda}{\operatorname{argmax}} \mathcal{L}_\Lambda, \quad (8)$$

where  $\sigma$  is a parameter of Gaussian priors and is set to 1.0 in later experiments. The partial derivatives of  $\mathcal{L}_\Lambda$  are as follows:

$$\begin{aligned} \frac{\partial \mathcal{L}_\Lambda}{\partial \lambda_k} = & \sum_{l=1}^L \left[ \sum_{\mathbf{s}} P_\Lambda(\mathbf{s} | \mathbf{w}^l, \mathbf{h}^l, p^l) F_k(\mathbf{w}^l, \mathbf{h}^l, \mathbf{s}) \right. \\ & \left. - \sum_{\mathbf{s}} P_\Lambda(\mathbf{s} | \mathbf{w}^l, \mathbf{h}^l) F_k(\mathbf{w}^l, \mathbf{h}^l, \mathbf{s}) \right] - \frac{1}{\sigma^2} \lambda_k. \end{aligned} \quad (9)$$

The model parameters can be calculated with the L-BFGS quasi-Newton method (Liu and Nocedal, 1989) using the objective function and its partial derivatives. While the partial derivatives contain summation over all the possible configurations of hidden variables, it can be calculated efficiently using belief propagation as explained in Section 2.2. This parameter estimation method is same to one used for Latent-Dynamic Conditional Random Field (Morency et al., 2007). Note that the objective function  $\mathcal{L}_\Lambda$  is not convex, and there is no guarantee for global optimality. The estimated model parameters depend on the initial values of the parameters, and the setting of the initial values of model parameters will be explained in Section 2.4.

## 2.4 Features

Table 1 shows the features used in this study. Features (a)–(h) in Table 1 are used as the node features (Equation (4)) for the  $i$ -th phrase, and features (A)–(E) are used as the edge features for the  $i$ -th and  $j$ -th phrases ( $j=h_i$ ). In Table 1,  $s_i$  denotes the hidden variable which represents the polarity of the dependency subtree whose root node is the  $i$ -th phrase,  $q_i$  denotes the prior polarity of the  $i$ -th phrase (explained later),  $r_i$  denotes the polarity reversal of the  $i$ -th phrase (explained later),  $m_i$  denotes the number of words in the  $i$ -th phrase,  $u_{i,k}$ ,  $b_{i,k}$ ,  $c_{i,k}$ ,  $f_{i,k}$  respectively denote the surface form, base form, coarse-grained part-of-speech (POS) tag,

Node Features	
a	$s_i$
b	$s_i \& q_i$
c	$s_i \& q_i \& r_i$
d	$s_i \& u_{i,1}, \dots, s_i \& u_{i,m_i}$
e	$s_i \& c_{i,1}, \dots, s_i \& c_{i,m_i}$
f	$s_i \& f_{i,1}, \dots, s_i \& f_{i,m_i}$
g	$s_i \& u_{i,1} \& u_{i,2}, \dots, s_i \& u_{i,m_i-1} \& u_{i,m_i}$
h	$s_i \& b_{i,1} \& b_{i,2}, \dots, s_i \& b_{i,m_i-1} \& b_{i,m_i}$
Edge Features	
A	$s_i \& s_j$
B	$s_i \& s_j \& r_j$
C	$s_i \& s_j \& r_j \& q_j$
D	$s_i \& s_j \& b_{i,1}, \dots, s_i \& s_j \& b_{i,m_i}$
E	$s_i \& s_j \& b_{j,1}, \dots, s_i \& s_j \& b_{j,m_j}$

Table 1: Features Used in This Study

fine-grained POS tag of the  $k$ -th word in the  $i$ -th phrase.

We used the morphological analysis system JUMAN and the dependency parser KNP<sup>2</sup> for processing Japanese data, and the POS tagger MXPOST (Ratnaparkhi, 1996) and the dependency parser MaltParser<sup>3</sup> for English data. KNP outputs phrase-based dependency trees, but MaltParser outputs word-based dependency trees, and we converted the word-based ones to phrase-based ones using simple heuristic rules explained in Appendix A.

The prior polarity of a phrase  $q_i \in \{+1, 0, -1\}$  is the innate sentiment polarity of a word contained in the phrase, which can be obtained from sentiment polarity dictionaries. We used sentiment polarity dictionaries made by Kobayashi et al. (2007) and Higashiyama et al. (2008)<sup>4</sup> for Japanese experiments (The resulting dictionary contains 6,974 positive expressions and 8,428 negative expressions), and a dictionary made by Wilson et al. (2005)<sup>5</sup> for English experiments (The dictionary contains 2,289 positive expressions and 4,143 negative expressions). When a phrase contains the words registered in the dictionaries, its prior polarity is set to the registered polarity, otherwise the prior polarity is set to 0. When a phrase contains multiple words in the dictionaries, the registered polarity of the last (nearest to the end

<sup>2</sup><http://nlp.kuee.kyoto-u.ac.jp/nl-resource/>

<sup>3</sup><http://maltparser.org/>

<sup>4</sup><http://cl.naist.jp/~inui/research/EM/sentiment-lexicon.html>

<sup>5</sup><http://www.cs.pitt.edu/mpqa/>

of the sentence) word is used.

The polarity reversal of a phrase  $r_i \in \{0, 1\}$  represents whether it reverses the polarities of other phrases (1) or not (0). We prepared polarity reversing word dictionaries, and the polarity reversal of a phrase is set to 1 if the phrase contains a word in the dictionaries, otherwise set to 0. We constructed polarity reversing word dictionaries which contain such words as *decrease* and *vanish* that reverse sentiment polarity. A Japanese polarity reversing word dictionary was constructed from an automatically constructed corpus, and the construction procedure is described in Appendix B (The dictionary contains 219 polarity reversing words). An English polarity reversing word dictionary was constructed from the General Inquirer dictionary<sup>6</sup> in the same way as Choi and Cardie (2008), by collecting words which belong to either NOTLW or DECREAS categories (The dictionary contains 121 polarity reversing words).

Choi and Cardie (2008) categorized polarity reversing words into two categories: function-word negators such as *not* and content-word negators such as *eliminate*. The polarity reversal of a phrase  $r_i$  explained above handles only the content-word negators, and function-word negators are handled in another way, since the scope of a function-word negator is generally limited to the phrase containing it in Japanese, and the number of function-word negators is small. The prior polarity  $q_i$  and the polarity reversal  $r_i$  of a phrase are changed to the following  $q'_i$  and  $r'_i$ , if the phrase contains a function-word negator (in Japanese) or if the phrase is modified by a function-word negator (in English):

$$q'_i = -q_i, \quad (10)$$

$$r'_i = 1 - r_i. \quad (11)$$

In this paper, unless otherwise noted, the word *polarity reversal* is used to indicate polarity reversing caused by content-word negators, and function-word negators are assumed to be applied to  $q_i$  and  $r_i$  in the above way beforehand.

As described in Section 2.3, there is no guarantee of global optimality for estimated parameters, since the objective function is not convex. In our

<sup>6</sup><http://www.wjh.harvard.edu/inquirer/>

preliminary experiments, L-BFGS often did not converge and classification accuracy was unstable when the initial values of parameters were randomly set. Therefore, in later experiments, we set the initial values in the following way. For the feature (A) in Table 1 in which  $s_i$  and  $s_j$  are equal, we set the initial parameter  $\lambda_i$  of the feature to a random number in  $[0.9, 1.1]$ , otherwise we set to a random number in  $[-0.1, 0.1]$ <sup>7</sup>. By setting such initial values, the initial model parameters have a property that two phrases with head-modifier relation tend to have the same polarity, which is intuitively reasonable.

### 3 Experiments

We conducted experiments of sentiment classification on four Japanese corpora and four English corpora.

#### 3.1 Data

We used four corpora for experiments of Japanese sentiment classification: the Automatically Constructed Polarity-tagged corpus (ACP) (Kaji and Kitsuregawa, 2006), the Kyoto University and NTT Blog corpus (KNB)<sup>8</sup>, the NTCIR Japanese opinion corpus (NTC-J) (Seki et al., 2007; Seki et al., 2008), the 50 Topics Evaluative Information corpus (50 Topics) (Nakagawa et al., 2008). The ACP corpus is an automatically constructed corpus from HTML documents on the Web using lexico-syntactic patterns and layout structures. The size of the corpus is large (it consists of 650,951 instances), and we used 1/100 of the whole corpus. The KNB corpus consists of Japanese blogs, and is manually annotated. The NTC-J corpus consists of Japanese newspaper articles. There are two NTCIR Japanese opinion corpora available, the NTCIR-6 corpus and the NTCIR-7 corpus; and we combined the two corpora. The 50 Topics corpus is collected from various pages on the Web, and is manually annotated.

We used four corpora for experiments of English sentiment classification: the Customer Review data

<sup>7</sup>The values of most learned parameters distributed between -1.0 and 1.0 in our preliminary experiments. Therefore, we decided to give values around the upper bound (1.0) and the mean (0.0) to the features in order to incorporate minimal prior knowledge into the model.

<sup>8</sup><http://nlp.kuee.kyoto-u.ac.jp/kuntt/>

(CR)<sup>9</sup>, the MPQA Opinion corpus (MPQA)<sup>10</sup>, the Movie Review Data (MR)<sup>11</sup>, and the NTCIR English opinion corpus (NTC-E) (Seki et al., 2007; Seki et al., 2008). The CR corpus consists of review articles about products such as digital cameras and cellular phones. There are two customer review datasets, the 5 products dataset and the 9 products dataset, and we combined the two datasets. In the MPQA corpus, sentiment polarities are attached not to sentences but expressions (sub-sentences), and we regarded the expressions as sentences and classified the polarities. There are two NTCIR English corpora available, the NTCIR-6 corpus and the NTCIR-7 corpus, and we combined the two corpora.

The statistical information of the corpora we used is shown in Table 2. We randomly split each corpus into 10 portions, and conducted 10-fold cross validation. Accuracy of sentiment classification was calculated as the number of correctly predicted labels (polarities) divided by the number of test examples.

### 3.2 Compared Methods

We compared our method to 6 baseline methods, and this section describes them. In the following,  $p_0 \in \{+1, -1\}$  denotes the major polarity in training data,  $\mathbf{H}_i$  denotes the set consisting of all the ancestor nodes of the  $i$ -th phrase in the dependency tree, and  $\text{sgn}(x)$  is defined as below:

$$\text{sgn}(x) = \begin{cases} +1 & (x > 0), \\ 0 & (x = 0), \\ -1 & (x < 0). \end{cases}$$

**Voting without Polarity Reversal** The polarity of a subjective sentence is decided by voting of each phrase’s prior polarity. In the case of a tie, the major polarity in the training data is adopted.

$$p = \text{sgn}\left(\sum_{i=1}^n q_i + 0.5p_0\right). \quad (12)$$

**Voting with Polarity Reversal** Same to *Voting without Polarity Reversal*, except that the polarities of phrases which have odd numbers of

reversal phrases in their ancestors are reversed before voting.

$$p = \text{sgn}\left(\sum_{i=1}^n q_i \prod_{j \in \mathbf{H}_i} (-1)^{r_j} + 0.5p_0\right). \quad (13)$$

**Rule** The polarity of a subjective sentence is deterministically decided basing on rules, by considering the sentiment polarities of dependency subtrees. The polarity of the dependency subtree whose root is the  $i$ -th phrase is decided by voting the prior polarity of the  $i$ -th phrase and the polarities of the dependency subtrees whose root nodes are the modifiers of the  $i$ -th phrase. The polarities of the modifiers are reversed if their head phrase has a reversal word. The decision rule is applied from leaf nodes in the dependency tree, and the polarity of the root node is decided at the last.

$$s_i = \text{sgn}\left(q_i + \sum_{j: h_j=i} s_j (-1)^{r_i}\right), \quad (14)$$

$$p = \text{sgn}(s_0 + 0.5p_0). \quad (15)$$

**Bag-of-Features with No Dictionaries** The polarity of a subjective sentence is classified using Support Vector Machines. Surface forms, base forms, coarse-grained POS tags and fine-grained POS tags of word unigrams and bigrams in the subjective sentence are used as features<sup>12</sup>. The second order polynomial kernel is used and the cost parameter  $C$  is set to 1.0. No prior polarity information (dictionary) is used.

**Bag-of-Features without Polarity Reversal** Same to *Bag-of-Features with No Dictionaries*, except that the voting result of prior polarities (one of positive, negative or tie) is also used as a feature.

**Bag-of-Features with Polarity Reversal** Same to *Bag-of-Features without Polarity Reversal*, except that the polarities of phrases which have

<sup>9</sup><http://www.cs.uic.edu/liub/FBS/sentiment-analysis.html>

<sup>10</sup><http://www.cs.pitt.edu/mpqa/>

<sup>11</sup><http://www.cs.cornell.edu/People/pabo/movie-review-data/>

<sup>12</sup>In experiments on English corpora, only the features of unigrams are used and those of bigrams are not used, since the bigram features decreased accuracies in our preliminary experiments as reported in previous work (Andreevskaia and Bergler, 2008).

Language	Corpus	Number of Instances	(Positive / Negative)
Japanese	ACP	6,510	(2,738 / 3,772)
	KNB	2,288	(1,423 / 865)
	NTC-J	3,485	(1,083 / 2,402)
	50 Topics	5,366	(3,175 / 2,191)
English	CR	3,772	(2,406 / 1,366)
	MPQA	10,624	(3,316 / 7,308)
	MR	10,662	(5,331 / 5,331)
	NTC-E	3,812	(1,226 / 2,586)

Table 2: Statistical Information of Corpora

Method	Japanese				English			
	ACP	KNB	NTC-J	50 Topics	CR	MPQA	MR	NTC-E
Voting-w/o Rev.	0.686	0.764	0.665	0.727	0.714	0.804	0.629	0.730
Voting-w/ Rev.	0.732	0.792	0.714	0.765	0.742	0.817	0.631	0.740
Rule	0.734	0.792	0.742	0.764	0.743	0.818	0.629	0.750
BoF-no Dic.	0.798	0.758	0.754	0.761	0.793	0.818	0.757	0.768
BoF-w/o Rev.	0.812	0.823	0.794	0.805	0.802	0.840	0.761	0.793
BoF-w/ Rev.	0.822	0.830	0.804	0.819	<b>0.814</b>	0.841	0.764	0.797
Tree-CRF	<b>0.846*</b>	<b>0.847*</b>	<b>0.826*</b>	<b>0.841*</b>	<b>0.814</b>	<b>0.861*</b>	<b>0.773*</b>	<b>0.804</b>

(\* indicates statistical significance at  $p < 0.05$ )

Table 3: Accuracy of Sentiment Classification

odd numbers of reversal phrases in their ancestors are reversed before voting.

**Tree-CRF** The proposed method based on dependency trees using CRFs, described in Section 2.

### 3.3 Experimental Results

The experimental results are shown in Table 3. The proposed method Tree-CRF obtained the best accuracies for all the four Japanese corpora and the four English corpora, and the differences against the second best methods were statistically significant ( $p < 0.05$ ) with the paired t-test for the six of the eight corpora. Tree-CRF performed better for the Japanese corpora than for the English corpora. For both the Voting methods and the Bag-of-Features methods, the methods with polarity reversal performed better than those without it<sup>13</sup>.

Both BoF-w/ Rev. and Tree-CRF use supervised machine learning and the same dictionaries (the

<sup>13</sup>The Japanese polarity reversing word dictionary was constructed from the ACP corpus as described in Appendix B, and it is not reasonable to compare the methods with and without polarity reversal on the ACP corpus. However, the tendency can be seen on the other 7 corpora.

prior polarity dictionaries and the polarity reversing word dictionaries), but the latter performed better than the former. Our error analysis showed that BoF-w/ Rev. was not robust for erroneous words in the prior polarity dictionaries. BoF-w/ Rev. uses the voting result of the prior polarities as a feature, and the feature is sensitive to the errors in the dictionary, while Tree-CRF uses several information as well as the prior polarities to decide the polarities of dependency subtrees, and was robust to the dictionary errors. We investigated the trained model parameters of Tree-CRF, and found that the features (E) in Table 1, in which the head and the modifier have opposite polarities and the head word is such as *protect* and *withdraw*, have large positive weights. Although these words were not included in the polarity reversing word dictionary, the property that these words reverse polarities of other words seems to be learned with the model.

## 4 Related Work

Various studies on sentiment classification have been conducted, and there are several methods pro-

posed for handling reversal of polarities. In this paper, our method was not directly compared with the other methods, since it is difficult to completely implement them or conduct experiments with exactly the same settings.

Choi and Cardie (2008) proposed a method to classify the sentiment polarity of a sentence basing on compositional semantics. In their method, the polarity of the whole sentence is determined from the prior polarities of the composing words by pre-defined rules, and the method differs from ours which uses the probabilistic model to handle interactions between hidden variables. Syntactic structures were used in the studies of Moilanen and Pulman (2007) and, Jia et al. (2009), but their methods are based on rules and supervised learning was not used to handle polarity reversal. As discussed in Section 1, Wilson et al. (2005) studied a bag-of-features based statistical sentiment classification method incorporating head-modifier relation.

Ikeda et al. (2008) proposed a machine learning approach to handle sentiment polarity reversal. For each word with prior polarity, whether the polarity is reversed or not is learned with a statistical learning algorithm using its surrounding words as features. The method can handle only words with prior polarities, and does not use syntactic dependency structures.

Conditional random fields with hidden variables have been studied so far for other tasks. Latent-Dynamic Conditional Random Fields (LDCRF) (Morency et al., 2007; Sun et al., 2008) are probabilistic models with hidden variables for sequential labeling, and belief propagation is used for inference. Our method is similar to the models, but there are several differences. In our method, only one variable which represents the polarity of the whole sentence is observable, and dependency relation among random variables is not a linear chain but a tree structure which is identical to the syntactic dependency.

## 5 Conclusion

In this paper, we presented a dependency tree-based method for sentiment classification using conditional random fields with hidden variables. In this method, the polarity of each dependency subtree

of a subjective sentence is represented by a hidden variable. The values of the hidden variables are calculated in consideration of interactions between variables whose nodes have head-modifier relation in the dependency tree. The value of the hidden variable of the root node is identified with the polarity of the whole sentence. Experimental results showed that the proposed method performs better for Japanese and English data than the baseline methods which represents subjective sentences as bag-of-features.

## Appendix

### A Rules for Converting Word Sequence to Phrase Sequence

Let  $v_1, \dots, v_N$  denote an English word sequence,  $y_i$  the part-of-speech of the  $i$ -th word, and  $z_i$  the head index of the  $i$ -th word. The word sequence was converted to a phrase sequence as follows, by applying rules which combine two adjacent words:

```

LT ≡ {“,(-LRB,-LSB,-LCB,CC}
RT ≡ {“,),,-,.,,;POS,-RRB,-RSB,-RCB-}
PP ≡ {IN,RP,TO,DT,PDT,PRP,WDT,WP,WPS,WRB}
NN ≡ {CD,FW,NN,NNP,NNPS,NNS,SYM,JJ}
do
  for  $i := 1$  to  $N - 1$ 
    if  $x_i$  and  $x_{i+1}$  are not yet combined  $\wedge$ 
      ( $x_i \in LT \vee$ 
        $x_{i+1} \in RT \vee$ 
       ( $(y_i = y_{i+1} \vee y_i = i + 1 \vee y_{i+1} = i) \wedge$ 
        ( $x_i \in PP \vee$ 
         ( $x_i \in NN \wedge x_{i+1} \in NN))))$ ) then
      Combine the words  $v_i$  and  $v_{i+1}$ 
  until No rules are applied

```

### B Construction of Japanese Polarity Reversing Word Dictionary

We constructed a Japanese polarity reversing word dictionary from the Automatically Constructed Polarity-tagged corpus (Kaji and Kitsuregawa, 2006). First, we collected sentences, each of which contains just one phrase having prior polarity, and the phrase modifies a phrase which modifies the root node. Among them, we selected sentences in which the prior polarity is not equal to the polarity of the whole sentence. We extracted all the words in the head phrase, and manually checked them whether they should be put into the dictionary or not. The rationale behind the procedure is that the prior polarity can be considered to be reversed by a certain word in the head phrase.



## References

- Alina Andreevskaia and Sabine Bergler. 2008. When Specialists and Generalists Work Together: Overcoming Domain Dependence in Sentiment Tagging. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 290–298.
- Soumen Chakrabarti. 2002. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan-Kaufman.
- Yejin Choi and Claire Cardie. 2008. Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 793–801.
- Masahiko Higashiyama, Kentaro Inui, and Yuji Matsumoto. 2008. Acquiring Noun Polarity Knowledge Using Selectional Preferences. In *Proceedings of the 14th Annual Meeting of the Association for Natural Language Processing*, pages 584–587. (in Japanese).
- Daisuke Ikeda, Hiroya Takamura, Lev-Arie Ratinov, and Manabu Okumura. 2008. Learning to Shift the Polarity of Words for Sentiment Classification. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 296–303.
- Lifeng Jia, Clement Yu, and Weiyi Meng. 2009. The Effect of Negation on Sentiment Analysis and Retrieval Effectiveness. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pages 1827–1830.
- Nobuhiro Kaji and Masaru Kitsuregawa. 2006. Automatic Construction of Polarity-Tagged Corpus from HTML Documents. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 452–459.
- Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. 2007. Opinion Mining from Web Documents: Extraction and Structurization. *Journal of the Japanese Society for Artificial Intelligence*, 22(2):227–238.
- Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(3):503–528.
- David J. C. MacKay. 2003. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.
- Karo Moilanen and Stephen Pulman. 2007. Sentiment Composition. In *Proceedings of the Recent Advances in Natural Language Processing International Conference*, pages 378–382.
- Louis-Philippe Morency, Ariadna Quattoni, and Trevor Darrell. 2007. Latent-Dynamic Discriminative Models for Continuous Gesture Recognition. In *Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.
- Tetsuji Nakagawa, Takuya Kawada, Kentaro Inui, and Sadao Kurohashi. 2008. Extracting Subjective and Objective Evaluative Expressions from the Web. In *Proceedings of the 2nd International Symposium on Universal Communication*.
- Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.
- 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*, pages 79–86.
- Adwait Ratnaparkhi. 1996. A Maximum Entropy Model for Part-of-Speech Tagging. In *Proceedings of the 1996 Conference on Empirical Methods in Natural Language Processing Conference*, pages 133–142.
- Yohei Seki, David Kirk Evans, Lun-Wei Ku, Hsin-Hsi Chen, Noriko Kando, and Chin-Yew Lin. 2007. Overview of Opinion Analysis Pilot Task at NTCIR-6. In *Proceedings of the 6th NTCIR Workshop*, pages 265–278.
- Yohei Seki, David Kirk Evans, Lun-Wei Ku, Le Sun, Hsin-Hsi Chen, and Noriko Kando. 2008. Overview of Multilingual Opinion Analysis Task at NTCIR-7. In *Proceedings of the 7th NTCIR Workshop*.
- Xu Sun, Louis-Philippe Morency, Daisuke Okanohara, and Jun’ichi Tsujii. 2008. Modeling Latent-Dynamic in Shallow Parsing: A Latent Conditional Model with Improved Inference. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 841–848.
- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of the 2005 Joint Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354.
- Livia Polanyi Annie Zaenen. 2004. Contextual Lexical Valence Shifters. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text*.