Abstract

In this paper, we propose a data-oriented method for inferring the emotion of a speaker conversing with a dialog system from the semantic content of an utterance. We first fully automatically obtain a huge collection of emotion-provoking event instances from the Web. With Japanese chosen as a target language, about 1.3 million emotion-provoking event instances are extracted using an emotion lexicon and lexical patterns. We then decompose the emotion classification task into two sub-steps: sentiment polarity classification (coarse-grained emotion classification), and emotion classification (fine-grained emotion classification). For each subtask, the collection of emotion-provoking event instances is used as labelled examples to train a classifier. The results of our experiments indicate that our method significantly outperforms the baseline method.

1 Introduction

Previous research into human-computer interaction has mostly focused on task-oriented dialogs, where the goal is considered to be to achieve a given task as precisely and efficiently as possible by exchanging information required for the task through dialog (Allen et al., 1994, etc.). More recent research (Foster, 2007; Tokuhisa and Terashima, 2006, etc.), on the other hand, has been providing evidence for the importance of the affective or emotional aspect in a wider range of dialogic contexts, which has been largely neglected in the context of task-oriented dialogs.

A dialog system may be expected to serve, for example, as an active listening partner of an elderly user living alone who sometimes wishes to have a chat. In such a context, the dialog system is expected to understand the user’s emotions and sympathize with the user. For example, given an utterance I traveled far to get to the shop, but it was closed from the user, if the system could infer the user’s emotion behind it, it would know that it would be appropriate to say That’s too bad or That’s really disappointing. It can be easily imagined that such affective behaviors of a dialog system would be beneficial not only for active listening but also for a wide variety of dialog purposes including even task-oriented dialogs.

To be capable of generating sympathetic responses, a dialog system needs a computational model that can infer the user’s emotion behind his/her utterance. There have been a range of studies for building a model for classifying a user’s emotions based on acoustic-prosodic features and facial expressions (Pantic and Rothkrantz, 2004, etc.). Such methods are, however, severely limited in that they tend to work well only when the user expresses his/her emotions by “exaggerated”

---

1Active listening is a specific communication skill, based on the work of psychologist Carl Rogers, which involves giving free and undivided attention to the speaker (Robertson, 2005).
prosodic or facial expressions. Furthermore, what is required in generating sympathetic responses is the identification of the user's emotion in a finer grain-size. For example, in contrast to the above example of disappointing, one may expect the response to My pet parrot died yesterday should be That's really sad, whereas the response to I may have forgotten to lock my house should be You're worried about that.

In this paper, we address the above issue of emotion classification in the context of human-computer dialog, and demonstrate that massive examples of emotion-provoking events can be extracted from the Web with a reasonable accuracy and those examples can be used to build a semantic content-based model for fine-grained emotion classification.

2 Related Work

Recently, several studies have reported about dialog systems that are capable of classifying emotions in a human-computer dialog (Batliner et al., 2004; Ang et al., 2002; Litman and Forbes-Riley, 2004; Rotaru et al., 2005). ITSPOKE is a tutoring dialog system, that can recognize the user’s emotion using acoustic-prosodic features and lexical features. However, the emotion classes are limited to Uncertain and Non-Uncertain because the purpose of ITSPOKE is to recognize the user’s problem or discomfort in a tutoring dialog. Our goal, on the other hand, is to classify the user’s emotions into more fine-grained emotion classes.

In a more general research context, while quite a few studies have been presented about opinion mining and sentiment analysis (Liu, 2006), research into fine-grained emotion classification has emerged only recently. There are two approaches commonly used in emotion classification: a rule-based approach and a statistical approach. Masum et al. (2007) and Chaumartin (2007) propose a rule-based approach to emotion classification. Chaumartin has developed a linguistic rule-based system, which classifies the emotions engendered by news headlines using the WordNet, SentiWordNet, and WordNet-Affect lexical resources. The system detects the sentiment polarity for each word in a news headline based on linguistic resources, and then attempts emotion classification by using rules based on its knowledge of sentence structures. The recall of this system is low, however, because of the limited coverage of the lexical resources. Regarding the statistical approach, Kozareva et al. (2007) apply the theory of (Hatzivassiloglou and McKeown, 1997) and (Tunney, 2002) to emotion classification and propose a method based on the co-occurrence distribution over content words and six emotion words (e.g. joy, fear). For example, birthday appears more often with joy, while war appears more often with fear. However, the accuracy achieved by their method is not practical in applications assumed in this paper. As we demonstrate in Section 4, our method significantly outperforms Kozareva’s method.

3 Emotion Classification

3.1 The basic idea

We consider the task of emotion classification as a classification problem where a given input sentence (a user’s utterance) is to be classified either into such 10 emotion classes as given later in Table 1 or as (neutral) if no emotion is involved in the input. Since it is a classification problem, the task should be approached straightforwardly in a vari-
of machine learning-based methods if a sufficient number of labelled examples were available. Our basic idea is to learn what emotion is typically provoked in what situation, from massive examples that can be collected from the Web. The development of this approach and its subsequent implementation has forced us to consider the following two issues.

First, we have to consider the quantity and accuracy of emotion-provoking examples to be collected. The process we use to collect emotion-provoking examples is illustrated in the upper half of Figure 1. For example, from the sentence *I was disappointed because the shop was closed and I'd traveled a long way to get there*, pulled from the Web, we learn that the clause *the shop was closed and I'd traveled a long way to get there* is an example of an event that provokes *disappointment*. In this paper, we refer to such an example as an emotion-provoking event and a collection of emotion-provoking events as an emotion-provoking event corpus (an EP corpus). Details are described in Section 3.2.

Second, assuming that an EP corpus can be obtained, the next issue is how to use it for our emotion classification task. We propose a method whereby an input utterance (sentence) is classified in two steps, sentiment polarity classification followed by fine-grained emotion classification as shown in the lower half of Figure 1. Details are given in Sections 3.3 and 3.4.

### 3.2 Building an EP corpus

We used ten emotions *happiness, pleasantness, relief, fear, sadness, disappointment, unpleasantness, loneliness, anxiety, anger* in our emotion classification experiment. First, we built a handcrafted lexicon of emotion words classified into the ten emotions. From the Japanese Evaluation Expression Dictionary created by Kobayashi et al. (2005), we identified 349 emotion words based on the definition of emotion words proposed by Teramura (1982). The distribution is shown in Table 1 with major examples.

We then went on to find sentences in the Web corpus that possibly contain emotion-provoking events. A subordinate clause was extracted as an emotion-provoking event instance if (a) it was subordinated to a matrix clause headed by an emotion word and (b) the relation between the subordinate and matrix clauses is marked by one of the following eight connectives: で,から,ため,て,の,が,こと,が. An example is given in Figure 2. In the sentence “突然雨が降り出したの（が）に（I was disappointed） (it suddenly started raining)”, the subordinate clause “突然雨が降り出した (it suddenly started raining)” modifies “に (that)” with the connective “の (that)”. In this case, therefore, the event mention “突然雨が降り出した (it suddenly started raining)” is learned as an event instance that provokes (disappointment).

Applying the emotion lexicon and the lexical patterns to the Japanese Web corpus (Kawahara and Kurohashi, 2006), which contains 500 million sentences, we were able to collect about 1.3 million events as causes of emotion. The distribution is shown in Table 2.

Tables 3 and 4 show the results of our evalu-
3.3 Sentiment Polarity Classification

Given the large collection of emotion-labelled examples, it may seem straightforward to develop a trainable model for emotion classification. Before moving on to emotion classification, however, it should be noted that a user’s input utterance may not involve any emotion. For example, if the user gives an utterance I have a lunch at the school cafeteria every day, it is not appropriate for the system to make any sympathetic response. In such a case, the user’s input should be classified as ⟨neutral⟩.

The classification between emotion-involved and neutral is not necessarily a simple problem, however, because we have not found yet any practical method for collecting examples of the class ⟨neutral⟩. We cannot rely on the analogy to the pattern-based method we have adopted to collect emotion-provoking events — there seems no reliable lexico-syntactic pattern for extracting neutral examples. Alternatively, if the majority of the sentences on the Web were neutral, one would simply use a set of randomly sampled sentences as labelled data for ⟨neutral⟩. This strategy, however, does not work because neutral sentences are not the majority in real Web texts. As an attempt, we collected 1000 sentences randomly from the Web and investigated their distribution of sentiment polarity. The results, shown in Table 5, revealed that the ratio of neutral events was unexpectedly low. These results indicate the difficulty of collecting neutral events from Web documents.

Taking this problem into account, we adopt a two-step approach, where we first classify a given input into three sentiment polarity classes, either positive, negative or neutral, and then classify only those judged positive or negative into our 10 fine-grained emotion classes. In the first step, i.e. sentiment polarity classification, we use only the positive and negative examples stored in the EP corpus and assume sentence to be neutral if the output of the classification model is near the decision boundary. There are additional advantages in this approach. First, it is generally known that performing fine-grained classification after coarse classification often provides good results particularly when the number of the classes is large. Second, in the context of dialog, a misunderstanding the user’s emotion at the sentiment polarity level would be a disaster. Imagine that the system says You must be happy when the user in fact feels sad. As we show in Section 4.2, such fatal errors can be reduced by taking the two-step approach.
 Various methods have already been proposed for sentiment polarity classification, ranging from the use of co-occurrence with typical positive and negative words (Turney, 2002) to bag of words (Pang et al., 2002) and dependency structure (Kudo and Matsumoto, 2004). Our sentiment polarity classification model is trained with SVMs (Vapnik, 1995), and the features are {1-gram, 2-gram, 3-gram} of words and the sentiment polarity of the words themselves. Figure 3 illustrates how the sentence “子供の教育の負担が増える (The cost of educating my child increases)” is encoded to a feature vector. Here we assume the sentiment polarity of the “子供 (child)” and “教育 (education)” are positive, while the “負担 (cost)” is negative. These polarity values are represented in parallel with the corresponding words, as shown in Figure 3. By expanding {1-gram, 2-gram, 3-gram} in this lattice representation, the following list of features are extracted: 子供 (child), Positive, 子供 (child)-の (of), Positive-の (of), 子供 (child)-の (of)-教育 (education), etc.. The polarity value of each word is defined in our sentiment polarity dictionary, which includes 1880 positive words and 2490 negative words. To create this dictionary, one annotator identified positive and negative words from the 50 thousand most frequent words sampled from the Web. Table 6 shows some examples.

### 3.4 Emotion Classification

For fine-grained emotion classification, we propose a k-nearest-neighbor approach (kNN) using the EP corpus.

Given an input utterance, the kNN model retrieves k-most similar labelled examples from the EP corpus. Given the input The restaurant was very far but it was closed as Figure 1, for example, the kNN model finds similar labelled examples, say, labelled example {the shop was closed and I’d traveled far to get there} in the EP corpus.
Two annotators were asked the following three test sets:

For fine-grained emotion classification, we used a three-class problem. Specifically, positive sentences tend to be classified as neutral. This method has the quality of the EP corpus described in Section 3.2.

In the experiments, the following two models were compared:

Baseline: The baseline model simulates the method proposed by (Kozareva et al., 2007). Given an input sentence, their model first estimates the pointwise mutual information (PMI) between each content word $cw_j$ included in the sentence and emotion expression $e \in \{anger, disgust, fear, joy, sadness, surprise\}$ by $PMI(e,cw) = \log \frac{hits(e,cw)}{\text{hits}(cw)}$, where $hits(x)$ is a hit count of word(s) x on a Web search engine. The model then calculates the score of each emotion class $E_i$ by summing the PMI scores between each content word $cw_j$ in the input and emotion expression $e_i$ corresponding to that emotion class: $score(E_i) = \sum_j PMI(e_i,cw_j)$. Finally, the model chooses the best scored emotion class as an output. For our experiments, we selected the following 10 emotion expressions:

For hit counts, we used the Google search engine.

### TestSet1 (1p, acceptable): One of the above two annotators was asked to annotate each positive or negative sentence in TestSet1 with all the emotions involved in it. The number of emotions for a positive sentence was 1.48 on average, and 2.47 for negative sentences. Table 10 lists some examples. In the experiments, a model’s answer was considered correct if it was identical with one of the labelled classes.

### TestSet2: For TestSet2, we used the results of our experiments for estimating the quality of the EP corpus described in Section 3.2.

In the experiments, the following two models were compared:

Baseline: The baseline model simulates the method proposed by (Kozareva et al., 2007). Given an input sentence, their model first estimates the pointwise mutual information (PMI) between each content word $cw_j$ included in the sentence and emotion expression $e \in \{anger, disgust, fear, joy, sadness, surprise\}$ by $PMI(e,cw) = \log \frac{hits(e,cw)}{\text{hits}(cw)}$, where $hits(x)$ is a hit count of word(s) x on a Web search engine. The model then calculates the score of each emotion class $E_i$ by summing the PMI scores between each content word $cw_j$ in the input and emotion expression $e_i$ corresponding to that emotion class: $score(E_i) = \sum_j PMI(e_i,cw_j)$. Finally, the model chooses the best scored emotion class as an output. For our experiments, we selected the following 10 emotion expressions:

For hit counts, we used the Google search engine.
An important observation from Figure 5 is that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.

Regarding the issue of single-step vs. two-step, Figure 5 indicates that the two-step models tended to outperform the single-step models for all the test set. A paired t-test for TestSet2, however, did not reach significance. So we next examined this issue in further detail.

As argued in Section 3.3, in the context of human-computer dialog, a misunderstanding of the user’s emotion at the level of sentiment polarity would lead to a serious problem, which we call a fatal error. On the other hand, misclassifying a case of (happiness) as, for example, (pleasantness) may well be tolerable. Table 11 shows the ratio of fatal errors for each model. For TestSet2, the single-step 10-NN model made fatal errors in 30% of cases, while the two-step 10-NN model in only 17%. This improvement is statistically significant (p<0.01).

5 Conclusion

In this paper, we have addressed the issue of emotion classification assuming its potential applications to be human-computer dialog system including active-listening dialog. We first automatically collected a huge collection, as many as 1.3M, of emotion-provoking event instances from the Web. We then decomposed the emotion classification task into two sub-steps: sentiment polarity classification and emotion classification. In sentiment polarity classification, we used the EP-corpus as training data. The results of the polarity classification experiment showed that word n-gram features alone are more or less sufficient to classify positive and negative sentences when a very large amount of training data is available. In the emotion classification experiments, on the other hand, the accuracy can be considered as a crucial advantage.
we adopted the k-nearest-neighbor (kNN) method. The results of the experiments showed that our method significantly outperformed the baseline method. The results also showed that our two-step emotion classification was effective for fine-grained emotion classification. Specifically, fatal errors were significantly reduced with sentiment polarity classification before fine-grained emotion classification.

For future work, we first need to examine other machine learning methods to see their advantages and disadvantages in our task. We also need an extensive improvement in identifying neutral sentences. Finally, we are planning to apply our model to the active-listening dialog system that our group has been developing and investigate its effects on the user’s behavior.

References


<table>
<thead>
<tr>
<th>Table 11: Fatal error rate in emotion classification experiments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
</tr>
<tr>
<td>TestSet1</td>
</tr>
<tr>
<td>TestSet2</td>
</tr>
</tbody>
</table>