

Exploitative-enterprise Diagnosis Based on Users' Values

Yingying Lao † Tomoki Aoyama † Dongli Han †

1. Introduction

Exploitative enterprises indicate the enterprises which make their employees to work under malicious working conditions. It is important for job seekers to estimate the possibility that a certain enterprise might be exploitative before starting their job-seeking activities with it. In order to investigate a company's reputation, job seekers generally use some websites where they can find a lot of reviews on the enterprise which have been posted by its employees on an anonymous basis [1]. However, the process of going through all the reviews could become quite time-consuming, especially when the total number of reviews gets large.

An existing resolution automates this process by extracting specific expressions from the review set, whereas neglects taking into consideration the job-seekers' values [2]. Values (or specifically speaking, sense of values) indicate each person's understanding of what is important, which could be similar among some people, or quite different among others for various reasons. Based on different values, job seekers could have different impressions on the same enterprise. For example, a job-seeker who wants to earn overtime money might neglect the description of overtime work intentionally.

In this paper, we propose a method to extract exploitative expressions from the review set of an enterprise, and estimate the degree of exploitation of it for job seekers based on their sense of values.

2. Our Method

Our system consists of two functions. Firstly, the system extracts exploitative expressions (e.g., *much overtime work*, *salary is cheap*) from the reviews which have been posted on a website [3]. Secondly, our system estimates the degree of exploitation of a certain enterprise by employing the exploitative expressions extracted in the first function, and the sense of values set up by the job-seekers in advance.

2.1 Preparation

In order to generate a dictionary to match and extract exploitative expressions from the reviews, we have conducted a questionnaire on *which expression do you think is appropriate for an exploitative enterprise?*. Thirty seven kinds of exploitative-expression candidates which have been manually collected from the web in advance were presented to 30 senior college students in the middle of job hunting. As a result of the questionnaire, 2 kinds of expressions have been filtered out as inappropriate expressions, 20 kinds of expressions have been determined as fixed exploitative-expressions, and 15 kinds of expressions have been selected as configurable expressions which will be used for value setting as described in Sub-section 2.2.

In addition, a weight has been assigned to each of the remaining 35 kinds of exploitative expressions. Fig. 1 shows some examples

of the dictionary entry.

entry word	→	evaluation word,	polarity,	weight
<i>overtime work</i>	→	<i>much</i> ,	1,	24
<i>overtime work</i>	→	<i>a few</i> ,	-1,	24
<i>salary</i>	→	<i>low</i> ,	1,	26
<i>salary</i>	→	<i>high</i> ,	-1,	26

Fig.1. Examples of exploitation-expression dictionary entry

The polarity in Fig. 1 is used to indicate the existence of negative expressions.

2.2 Setting Values

A selection-form containing all the 15 exploitative expressions will be shown to users before entering the main system. Users could skip this step if they don't have any particular sense of values, or select a combination of exploitative expressions to represent their own sense of values.

2.3 Review Collection

After setting the sense of values, our system acquires the target web page for a specific enterprise, and parses the extracted reviews using Cabocha, a Japanese syntactic analysis tool [4].

2.4 Exploitative-expression Extraction

Based on the results from the Syntactic Analysis of the reviews, pairs of exploitative expression (entry word → evaluation word, polarity) are extracted in four manners using the exploitative-expression dictionary described in Sub-section 2.1. Table 1 shows details of the four extraction methods.

Table 1. Extraction methods

Extraction Method	Description	Example
Dependency relation	Dependency relation exists between entry word and evaluation word	(<i>overtime work</i> → <i>much</i> , 1) (<i>overtime work</i> → <i>less</i> , -1)
Reversed dependency relation	Reversed dependency relation exists between entry word and evaluation word	(<i>much</i> → <i>overtime work</i> , 1)
Dependency relation + Keyword	Particular keywords appear around the Dependency relation	(<i>Allowance</i> → <i>none</i> , 1) + "extra job"
Fixed expression	Some Fixed expressions are also employed when extraction is difficult using the above three methods	<i>The enterprise culture that it is hard to say what you want to say.</i>

† Nihon University

2.5 Calculation of the degree of exploitation

The degree of exploitation of an enterprise is calculated with the following formulas.

$$BID_j = \sum_{i=0}^{NT} (S_BP_i \times \frac{N_BP_i}{TN_j}) \quad (1)$$

$$S_BP_i = \sum_{i=0}^{NT} (BP_i \times W_i) \quad (2)$$

BID_j indicates the degree of exploitation of the j th enterprise. N_BP_i is the number of reviews containing the i th exploitative expression, and TN_j is the total number of the j th enterprise's reviews. BP_i in formula (2) is the total frequency of appearance of the i th exploitative expression in all the reviews that have been extracted for the j th enterprise. W_i stands for the weight of the i th exploitative expression where i varies from 1 to NT . NT here indicates the total number of exploitative expressions which is 35 as described in Sub-section 2.1. If a user has set up her or his own sense of values by choosing some kinds of exploitative expressions in Sub-section 2.2, the system will ignore the corresponding exploitative expressions when calculating the degree of exploitation.

3. Evaluation

We have conducted some experiments to confirm the effectiveness of our approaches.

3.1 Evaluation on the accuracy of exploitative-expression

We have collected 500 reviews for 5 enterprises randomly, and extracted all the exploitative expressions first by handcraft, and then by using our system. The Recall is 76.9% and the Precision is 88.2% showing the usefulness of the exploitative-expression dictionary and the effectiveness of our extraction methods.

3.2 Evaluation on the usefulness of considering sense of values

We have two goals in this experiment. The first goal is to examine whether setting up values could help the system catch a closer impression to the job seeker about an enterprise. In order to make this point clear, we have compared three situations: manual judgement by subjects (job seekers), system judgement considering subjects' values, and system judgement without considering subjects' values. Here, as it's difficult to set absolute criteria on whether an enterprise is exploitative or not, we calculate the exploitative degrees for two enterprises and evaluate them in a comparative manner.

The second goal is to confirm the effectiveness of the proposed calculation method for the degree of exploitation. Here, two baseline methods have been used to compare with the proposed method.

$$\text{Baseline 1: } BID_j = \frac{NB}{TN_j} \quad (3)$$

$$\text{Baseline 2: } BID_j = \frac{\sum_{i=0}^{NT} \delta_i}{NT} \quad (4)$$

NB in formula (3) is the number of reviews containing at least one exploitative expression. δ_i in formula (4) is set to be 1 when BP_i is greater than 0, and 0 when BP_i is equal to 0.

Two job-seekers have cooperated with our research as subjects. Each of them is asked to select two enterprises (Enterprise A and Enterprise B) from a set of enterprise candidates first, then read the reviews of each enterprise freely, and finally determine which enterprise she or he thinks is more exploitative (e.g., $A > B$ means Enterprise A is more exploitative than B). We regard this manual result as the correct answer and compare it with those of the system. Table 2 shows the results of the system with the first subject in which case the correct answer is $A > B$.

Table 2. Results of the system with the first subject

	a(No Values considered)	a'(Values considered)	b(No Values considered)	b'(Values considered)
Baseline 1	0.09	0.08	0.041	0.020
Baseline 2	0.2	0.17	0.086	0.057
Proposed method	9.09	8.78	2.237	0.959

As we can see from Table 2, although the same results (*i.e.*, $a > b$) as manual have been observed in all situations, the gap widens when users' values are considered which indicates the usefulness of considering values. Furthermore, $(a'/b')/(a/b)$ is 1.78, 1.29, and 2.25 for Baseline 1, Baseline 2, and the proposed method respectively, showing the superiority of the proposed calculation method in estimating exploitative degrees.

Table 3. Results of the system with the second subject

	c(No Values considered)	c'(Values considered)	d(No Values considered)	d'(Values considered)
Baseline 1	0.020	0.010	0.051	0.040
Baseline 2	0.057	0.028	0.143	0.114
Proposed method	0.556	0.475	2.191	1.707

Table 3 shows the results of the system with the second subject, where the correct answer is $C > D$. Here, $(c'/d')/(c/d)$ is 0.638 for baseline 1 and 0.616 for baseline 2 respectively, indicating the helplessness of sense of values while adapted to the baselines. On the other hand, we get 1.096 for our proposed method, indicating the system is approaching the human judgment although c' is still less than d' .

4. Ending

In this paper, we have proposed a method to extract exploitative expressions from the review set of an enterprise, and estimate the degree of exploitation of it for job seekers based on their sense of values. Experimental results have shown the effectiveness of our approach.

References

- [1] Career connection. <https://careerconnection.jp/review/16307/Vorkers>. <https://www.vorkers.com/>
- [2] Takaaki Hishida, Keisuke Sumigama, Masataka Endo, Yuichi Nakashima, and Tetsuro Miura, "An extraction method of the features of the company by using reviews," AIT Associated Repository of Academic Resources. Vol. 17,(2015), pp.97–100.
- [3] Company's diagnostic. <https://en-hyouban.com/>
- [4] Cabochoa. <https://taku910.github.io/cabochoa/>