Pattern Mining Approach to Unsupervised Definition Extraction

Yulan Yan  Chikara Hashimoto  Kentaro Torisawa
NICT Universal Communication Research Institute, Information Analysis Laboratory

1 Introduction

Definition sentences are useful sources for many NLP tasks, such as taxonomic and non-taxonomic Relation Extraction [7], Question Answering [1] and Paraphrase Acquisition [5]. Definitions of terms are also among the most common types of information users search for on the Web [8].

A great deal of automatic definition extraction methods are supervised, using manually crafted or semi-automatically learned lexico-syntactic patterns.  Patterns are either very simple sequences of words (e.g. English pattern “NP is a”[9], Chinese pattern “NP指的是”[10], Japanese pattern “NPとは*である。”[5]) or more complex sequences of words, parts of speech and chunks. However definition sentences occur in highly variable representation styles, and the most frequent definitional pattern “NP is a” is inherently very noisy. Also current approaches using manually created training data to extract definitions suffer from high labor cost.

In this paper, we propose an unsupervised method to extract definitions from the Web by automatically generating highly variable definition patterns. A training dataset $D_{all}$ which consists of two datasets, $D_d$ and $D_{nd}$, is constructed automatically. $D_d$ consists of the first sentence of each Wikipedia article. $D_{nd}$ is randomly sampled Web sentences. From $D_{all}$, a wide range of highly reliable definition patterns are generated automatically. A SVM classifier is trained on $D_{all}$ and then used to automatically extract definitions from a large Web data. The method is applied to English, Chinese and Japanese definition extraction. Experimental results show that our method is effective to extract multi-lingual definition sentences with low costs.

2 Related Work

A great deal of work is concerned with definition extraction. The majority of these approaches are supervised and language dependent, using lexico-syntactic patterns which are manually crafted or semi-automatically learned [7, 9, 4, 5, 10]. Only few papers try to cope with the generality of patterns and domains in real-world large corpora (like the Web). [1] proposed the use of probabilistic lexico-syntactic patterns, called soft patterns, to model definitions. The authors described a soft matching model based on a n-gram language model. [7] proposed a supervised method which learns word lattices to model textual definitions from an annotated dataset with complicated definition structure. Sentences in the training set are generalized to subsequence patterns which are then clustered. For each cluster, a word lattice is created to model a type of definition. Unlike these methods, our proposed method requires only the Wikipedia articles and Web texts of a target language.

3 Proposed Method

Our proposed method starts from an automatic construction of two datasets, a definition dataset $D_d$ and a non-definition dataset $D_{nd}$. Then from $D_d$ and $D_{nd}$, n-gram patterns, subsequence patterns and dependency subtrees are automatically generated as definition and non-definition patterns. Finally a SVM classifier is trained and used to extract definitions from Web data.

3.1 Dataset Construction

We build $D_d$ from Wikipedia articles by collecting each article’s first sentence. The title of the article is regarded as the target term and replaced with “<term>” in the definition sentence. We randomly sample Web sentences from large Web corpora to build $D_{nd}$. Take the case of English definition extraction. From English Wikipedia, we obtained 2,439,257 definition sentences as $D_d$ after removing first sentences of articles such as category, template, list, and so on. Six million English sentences are randomly sampled from a Web corpus ClueWeb09 as $D_{nd}$. ClueWeb09 is a Web corpus which consists of about 1 billion Web pages in ten languages that were collected in January and February 2009. For $D_{nd}$, we regard all the noun phrases as defined term candidates. For each non-definition sentence, we iteratively choose a noun phrase and replace it with “<term>” to derive a new sentence.

3.2 Pattern Generation

Given a definition dataset $D_d$ and a non-definition dataset $D_{nd}$, our method automatically generates definition patterns, such as “<term> is a” and “<term>とは*である。” which most of previous methods had to create manually. We assume that definition patterns are frequent in $D_d$ but are infrequent in $D_{nd}$, and non-definition patterns are frequent in $D_{nd}$ but are infrequent in $D_d$.

Our method generates three types of definition and non-definition patterns including n-gram patterns, subsequence patterns and dependency subtrees automatically by capturing significant differences between $D_d$ and $D_{nd}$. To mine definition and non-definition patterns, frequent patterns are generated from each dataset and the support $(supp)$ of a pattern $\phi$ in a dataset $D$ is calculated as follows [3]:

$$supp(\phi, D) := \frac{freq(\phi, D)}{|D|}$$

1http://lemurproject.org/clueweb09.php/
We generate definition and non-definition n-grams from \( D_{all} \). Frequent n-grams are collected from each dataset. A support threshold \( s_n \) and a minimum growth rate \( g_n \) are given to find all definition patterns which satisfy \( \text{supp}(\phi, D_d) \geq s_n \) and \( \text{growth}_{D_d \rightarrow D_d}(\phi) \geq g_n \), and all non-definition patterns which satisfy \( \text{supp}(\phi, D_{nd}) \geq s_n \) and \( \text{growth}_{D_d \rightarrow D_{nd}}(\phi) \geq g_n \). Thresholds \( s_n \) and \( g_n \) are set up for the following subsequence pattern generation and dimensionality reduction of SVM classification. Examples of English and Japanese n-gram patterns are shown in Table 1. We omit Chinese examples here for limited space.

### Subsequence Patterns

Subsequence patterns are combinations of ordered n-grams. Generating subsequence patterns using all the n-grams in \( D_{all} \) is very time consuming and will generate a huge number of subsequences. Therefore, we generate subsequences that consist of definition and non-definition n-gram patterns obtained from \( D_d \) and \( D_{nd} \). We take the sentence “AIG, led by Ken Ham, is one of the largest YEC organizations.” as an example.

Definition subsequence and non-definition subsequence patterns are obtained by giving support threshold \( s_p \) and growth rate threshold \( g_p \) values in the same way as n-gram patterns and subsequence patterns. The above figure shows two definition dependency subtree examples. t2 is a Japanese definition dependency subtree generated from a subsequence pattern “<term> とは、*である*” which was “*. For each word in the subsequence pattern (in blue font), we label its node as the word itself. For each other word (in blue font) in the subtree, we label the node as its part of speech.

From \( D_{all} \), definition and non-definition dependency subtrees are obtained by giving support threshold \( s_l \) and growth rate threshold \( g_l \) values in the same way as n-gram patterns and subsequence patterns. The above figure shows two definition dependency subtree examples. t2 is a Japanese definition dependency subtree generated from a subsequence pattern “<term> とは、*である*” which was “*. For each word in the subsequence pattern (in blue font), we label its node as the word itself. For each other word (in blue font) in the subtree, we label the node as its part of speech.

### Dependency Subtree Patterns

Dependency subtree patterns provide two types of information that n-gram patterns and subsequence patterns do not: dependency between words and part-of-speech of each word. The former is useful for distinguishing two sentences that have different noun phrases as <term> such as \( s_1 \) and \( s_2 \) below. \( s_1 \) and \( s_2 \) are derived from the same original sentence “AIG, led by Ken Ham, is one of the largest YEC organizations.”. As shown, the dependency subtree of sentence \( s_1 \) and \( s_2 \) are different. Intuitively, the subtree in \( s_1 \) is more likely to be a definition dependency subtree.
The latter information is useful to preserve important information which is omitted in n-gram patterns and subsequence patterns. For instance, s3 and s4 shown above match the same subsequence pattern “<term> is * which was *”. s3 is a definition but s4 is a non-definition. According to their dependency structure, the first “*” matches an objective noun phrase for s3, but an adjective phrase in s4. The dependency subtree in s3 is a definition dependency subtree (t1 shown above) but the dependency subtree in s4 is obviously not a definition subtree. Dependency subtree patterns can distinguish this difference, which n-gram patterns and subsequence patterns cannot.

3.3 Definition Classifier
A definition classifier is trained on the constructed $D_{all}$ with all the definition and non-definition patterns we generated. The classifier is applied to extract definitions from a Web corpus. For a Web sentence with more than one target term candidates, the classifier assigns a score for each candidate. The one with the highest score is taken as the target term.

4 Experimental Setting
In this paper, our claims are threefold:

- The performance of our unsupervised method is competitive to well-known supervised methods, with much less cost.
- All types of patterns that we propose contribute to the task.
- Our method is language independent.

Besides English, we also apply our method to Japanese and Chinese definition extraction. The training data for Japanese and Chinese are prepared in the same way as we did for English (Section 3.1). As shown in Table 4, the number of non-definition samples is 2 to 3 times the number of definition samples.

<table>
<thead>
<tr>
<th>Language</th>
<th>Positive</th>
<th>Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>2,439,257</td>
<td>5,000,000</td>
</tr>
<tr>
<td>JA</td>
<td>768,429</td>
<td>1,500,000</td>
</tr>
<tr>
<td>CH</td>
<td>310,072</td>
<td>600,000</td>
</tr>
</tbody>
</table>

Support and growth rate thresholds are turned off.

4.1 Proposed Method vs. Previous Methods
To compare to previous methods, we test our English definition classifier on an existing dataset\(^3\). It is a corpus of 4,619 Wikipedia sentences, containing 1,908 definition and 2,711 non-definition sentences. The former is a random selection of the first sentences of Wikipedia articles and the latter was obtained by extracting from the same Wikipedia articles sentences in which the page title occurs.

Our English definition classifier trained on $D_{all}$ is used to classify sentences in this dataset. Table 6 shows the results. “WCL” is the method proposed by [7]. [7] built this dataset and used it for both training and testing with 10-fold cross validation. [7] also implemented a baseline method denoted as “Star patterns” and [1]’s method denoted as “Bigrams” on the same datasets. “WCL” showed very high precision (P) (around 99%), higher than our proposed method (89.16%). However, our method achieves a much higher recall (R) (93.54% vs. 42.09%). Our proposed method achieves 91.30% in terms of F-measure (F), and the highest accuracy (A) 91.43%. Thus our method shows the best overall performance. Moreover, the “WCL” method is a supervised method which depends strongly on the annotated definition structure of their training data. Our method is an unsupervised method which uses automatically constructed training data.

4.2 Ablation Test
We conduct ablation tests to evaluate the contributions of different types of patterns. Three English definition classifiers are built on $D_{all}$:

- #1: uses only n-gram patterns as features.
- #2: uses n-gram patterns and subsequence patterns as features.
- #3: uses all the patterns we proposed.

Table 7 shows the performance of three classifiers on [7]’s dataset. The results show that each type of patterns contributes to the task. After adding dependency subtree patterns, we achieve lower precision but higher recall. The overall performance is the best with all types of patterns.

\(^2\)http://svmlight.joachims.org/

\(^3\)http://lcl.uniroma1.it/wcl
Table 7: Ablation test.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>87.87</td>
<td>85.64</td>
<td>86.74</td>
<td>89.69</td>
</tr>
<tr>
<td>#2</td>
<td>91.87</td>
<td>85.46</td>
<td>88.55</td>
<td>90.89</td>
</tr>
<tr>
<td>#3</td>
<td>89.16</td>
<td>93.54</td>
<td>91.30</td>
<td>93.46</td>
</tr>
</tbody>
</table>

Table 8: Performance on different languages.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>99.62</td>
<td>99.69</td>
<td>99.65</td>
<td>99.77</td>
</tr>
<tr>
<td>JA</td>
<td>98.32</td>
<td>94.83</td>
<td>96.54</td>
<td>98.25</td>
</tr>
<tr>
<td>CH</td>
<td>98.30</td>
<td>94.95</td>
<td>96.60</td>
<td>98.27</td>
</tr>
</tbody>
</table>

4.3 Evaluation of Language Independence

We apply our method to English, Japanese, and Chinese definition extraction to examine the independence of our method. Experiments are performed on the constructed training data $D_{all}$ for each language with 10-fold cross validation.

The results are shown in Table 8. Similar performance is observed for Japanese and Chinese as English, although training samples for Japanese and Chinese are fewer than those for English. All the systems perform well on $D_{all}$. One may wonder why the performance is much better than on [7]’s dataset. We suspect that one of the main reason is the selection of negative samples. [7] used Wikipedia sentences other than the first sentences of a Wikipedia article as negative examples. On the other hand, we use arbitrary Web sentences as negative examples. Therefore, another observation obtained from the results is that Web sentences can be more easily classified into non-definitions than Wikipedia non-definition sentences. Examples of obtained definitions are shown in Table 9.

5 Conclusion

In this paper, we propose an unsupervised method to extract definitions from the Web. A SVM classifier is trained on two automatically constructed datasets and applied to extract definitions from the Web data. Finally, we conclude that:

- From automatically constructed training datasets, $D_a$ and $D_{ad}$, a wide range of highly reliable definitional patterns can be generated automatically.

- Our method is a language independent method, as our experimental results showed: our method is effective to extract definition sentences of English, Japanese, and Chinese from the Web.

- Our proposed unsupervised method is competitive with the state-of-the-art supervised method.

References


