2014 IEEE International Conference on Data Mining Workshop

Automatic Learning Common Definitional Patterns from Multi-domain Wikipedia Pages

Jingsong Zhang
Department of CSE
Shanghai Jiao Tong University
Shanghai, China
Email: jasun_zhang@163.com

Yinglin Wang*
Department of CST
Shanghai University of Finance and Economics
Shanghai, China
Email: yinglin.wang@outlook.com

Dingyu Yang
Department of CSE
Shanghai Jiao Tong University
Shanghai, China
Email: yangdingyu8686@sjtu.edu.cn

***Abstract*—Automatic definition extraction has attracted wide interest in NLP domain and knowledge-based applications. One primary task of definition extraction is mining patterns from definitional sentences. Existing extraction methods of definitional patterns, either focus on manual extraction by intuition or observation, or aim to mine intricate definitional patterns by automatic extraction methods. The manual method requires large human resources to identify the definitional patterns because of diverse lexico-syntactic structures. It inevitable suffers poor behavior especially the extraction from cross-domain corpora. The latter method mainly considers the precision in definition extraction, which is at the cost of decreasing the recall of definitions. Both of them are unsuitable for cross-domain definition extraction. To address those issues, this paper proposes a solution to perform the automatic extraction of definitional patterns from multi-domain definitional sentences of Wikipedia. Our method FIND-SS is modified based on FIND-S algorithm and solves the definition extraction problems of cross-domain corpora. Find-SS adopts a "the more similar the higher priority" scheme to improve the learning performance. It can accommodate some noisy information and does not require any pattern seeds for pattern learning. The experimental results indicate that our scenario is significantly superior to previous method.**

***Keywords*—definition extraction; definitional pattern; FIND-S algorithm; similarity priority; frequent pattern**

I. Introduction

Entity definitions can be obtained to consist of dictionaries and domain glossaries for supporting the knowledge-based applications. However, manually extracting definitions requires the cooperative effort of many experts from multiple fields, which is laborious and time consuming. Furthermore, it is not feasible to manually extract the new definitions which are emerging constantly in the Web. Hence, automatically mining the complete and latest definitions from Web corpora is a significant task for constructing and updating glossaries. Automatic definition extraction is a useful tool not only in dictionary construction [1], but in other domains such as E-Learning [2], [3], question answering [4] and ontology engineering [5]–[7], taxonomy learning [8], [9], semantic predicate [10], etc. Currently, the automatic extraction of definitions from textual data has become a common research topic in several domains of Natural Language Processing (NLP) [11].

Much of the current literature focuses on employing a few lexico-syntactic patterns for definition extraction. The patterns can be directly obtained by intuition or observation from positive definitions. Some patterns rely on a few existing patterns from candidate definition extraction. English definitional patterns, such as *is, is a, refers to, is defined as, called,* are rather limited (about 10 types). As above sequences of words, especially copula *is*, occurs in both definitional and non-definitional sentences in a high probability. It suffers from both low recall and low precision if we rely only on such a few patterns for extraction. Therefore, it is necessary to mine more common patterns for preliminary extraction.

Some researchers tried to mine more complex lexico-syntactic patterns, part of speech (POS) and chunks. For instance, "$In\ *,\ a < definiendum > is\ a\ *$" [12], "$\{is|are\}[adverb]\{called|known\ as|defined\ as\}\{concept\}$" [13], "$NP\ (...)\ are/is\ NP-INS$" [14], etc. Using only these complex patterns to implement the definition extraction can obtain a higher precision. Nonetheless, it suffers from a lower recall and the patterns on positive definitions are very likely to be overfitting. Therefore, it is essential to find more common patterns for generating the candidate definitions.

Considering the problems, utilizing the common patterns extracts the candidate definitions from the textual data and then employing the complex patterns obtained by existing WCL (Word-class lattices) method identifies the positive definitions from the candidates, which is an available and attractive method. However, how to automatically learn sufficient common patterns is a major impediment to the definition extraction. To address the challenge, a similarity-based FIND-S algorithm, namely FIND-SS is presented to perform the automatic extraction of definitional patterns. First, definitional sentences (training examples) are formalized to a series of vectors that contain only "1" and "0" by n-gram and a set of string sequences. secondly, each two most similar sentence victors are generalized by FIND-S till all similar vectors are traversed. Thirdly, some noisy vectors are removed by a given *support*. Finally, the learner outputs the target vectors and their corresponding string sequences. It is the first report on Chinese definitional pattern extraction. About 8000 definitional and non-definitional sentences from Wikipedia articles of eight fields are collected for training and test. Our experimental results indicate that our techenque is superior to the state-of-art method.

The rest of this paper is organized as follows: Section 2 reviews the related work. Following that, in Section 3, the proposed technique is presented. Section 4 introduces the

*Corresponding author.

978-1-4799-4274-9/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDMW.2014.107

experiments and section 5 makes a short conclusion to the work and discusses our future work.

II. Related Work

Existing research on definition and candidate definition extraction mainly used the lexico-syntactic patterns taking into consideration typically POS or location structure. Przepiórkowski Adam et al [14] used 5 Bulgarian, 5 Czech and 7 Polish patterns for definition extraction. Furthermore, the researchers of [14] also evaluated the patterns in Polish [15]. Bing Liu et al. [13] exhibited 7 English patterns for identifying the definitions from the Web. In Chinese, Jingsong zhang et al. [5] used only 4 patterns and FangYie Leu et al. [16] employed 9 common and 4 domain patterns. Note that above researchers directly and simply used a few patterns for definition extraction. In comparison, Borg Claudia reported a further extension on this methodology. He [17] utilized 6 types of English patterns for candidate definition extraction and then learned the complex patterns from the candidate definitions. Soon afterwards, Borg Claudia et al. [18] employed 10 types of patterns for candidate definition extraction.

Also some literature focused on the Machine Learning (ML) method for definition extraction. Claudia Borg et al. [2], [17], [18] described the Genetic Algorithms (GA) and Genetic Programming (GP) for discovering the grammar patterns of definitions. Although the approach can discover some complex patterns, they also used simple and limited patterns as the seed patterns for candidate definition extraction [19]. It is encouraging that Roberto Navigli et al. [12], [19] presented a lattice-based approach to extract the definition and hypernym. It directly mined the complex definitional patterns from the positive definitions. A serious star patterns[1] were acquired and got 86.74% precision, but only 66.14% recall. We can note that previous researchers have done some inspiring work in definition extraction purpose. Nevertheless, there still leaves some room for improving the performance. which is learning more common definitional patterns for in-depth distinguishing.

III. Similarity-based FIND-S algorithm

A. Problem formulation

As a convention, let us consider the classic example of concept learning: *"days on which my friend Aldo enjoys his favorite water sport."* The attribute EnjoySport indicates whether or not Aldo enjoys his favorite water sport on this day. The task is learning to predict the value of EnjoySport for an arbitrary day, based on the values of its other attributes [20]. The goal of this task is to find a hypothesis h such that $h(x) = c(x)$ for each x in X, where X denotes the set of instances; $c(x)$ is the target concept value.

To smooth the following presentation, we refer to the definition of *more general than* relationship between hypotheses [20]. Let h_j and h_k be boolean-valued functions defined over X. Then h_j is *more_general_than_or_equal_to* h_k (written $h_j \geq_g h_k$) if and only if $(\forall x \in X)[(h_k(x) = 1) \rightarrow (h_j(x) = 1)]$.

[1]A star pattern, for example, a sequential string "In *, a <TARGET> is a *" can correspond with a original sentence "In arts, a chiaroscuro is a monochrome picture".

The FIND-S algorithm is an approach that search for an acceptable hypothesis by the *more_general_than* partial ordering. It is available if and only if (1) the correct target concept is in hypothesis H, and (2) every training example is positive. However, some negative examples (called noise), in fact, are inevitable in real-world corpora. As a consequence, the training examples including some noise will mislead FIND-S to obtain an error hypothesis. To illustrate the problem of our discussion, we regard the same task of learning the target concept days on which my friend Aldo enjoys his favorite water sport. Table 1 gives another example set.

As above mentioned example, can we find a maximally specific hypothesis from Table 1 training examples by FIND-S algorithm? If we implement the FIND-S, we have to face two problems:

(1) The number is quite small (only 3 and 4) both the count of positive (Yes) and negative (No) examples in the 4th group (ID is 4). If the approach scans each example including the small probability examples, the hypothesis may be rapidly converge to an over generalization hypothesis that is expression $<?,?,?,?,?,? >$, where ? is represented as any acceptable value for the corresponding attribute.

(2) The count of positive examples (ID is 7) is small (only 10), but the counterpart is big (up to 69).

To simplify the problem we assume the ID 1, 2 and 3 are all positive. We still use the FIND-S for finding a maximally specific hypothesis as follows:

1) $h \leftarrow < \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing >$
   // initialize h to the most specific hypothesis
2) $h \leftarrow < Sunny, Warm, Normal, Strong, Warm, Same >$
   // ID 1
3) $h \leftarrow < Sunny, Warm, ?, Strong, Warm, Same >$
   // ID 2
4) $h \leftarrow <?,?,?,?,?,? >$ // ID 3

Before accomplishing the scan of examples, the FIND-S has to terminate its crawling because the 4th step (ID is 3) leads the object hypothesis to an over generalization hypothesis h. However, it is a most and much more generalization hypothesis that no day is negative example. The hypothesis that we obtain by FIND-S is obvious too general to fit in the other training examples and next predicting. This is the third problem we have to face:

(3) The hypothesis is generalized to the most generalization hypothesis, i.e. $<?,?,?,?,?,? >$.

These three problems always occur in real-world learning applications, such as disease diagnosis, weather forecast, definition extraction, etc. The measurements of *support* and *vote* are introduced to handle respectively the first two problems. In addition, we propose FIND-SS approach to tackle the 3rd problem. To further quest above issues, we define some terms as follows.

Definition 1. A 1-meta definitional pattern is a pattern which includes only a string sequence, such as $\cdots$*is*$\cdots$, $\cdots$*is a*$\cdots$, $\cdots$*refer to*$\cdots$, etc.

Definition 2. A 2-meta definitional pattern is a pattern which includes two unconnected string sequences, such as $\cdots a \cdots$*is a* $\cdots$, *The*$\cdots$*is*$\cdots$, $\cdots$*is*$\cdots$*used for*$\cdots$, etc.

TABLE 1. THE NUMBER OF DAYS WITH TRAINING EXAMPLES FOR THE TARGET CONCEPT ENJOYSPORT.

								Count	
ID	Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	Yes	No
1	1-35	Sunny	Warm	Normal	Strong	Warm	Same	30	5
2	36-73	Sunny	Warm	High	Strong	Warm	Same	35	3
3	74-97	Cloudy	Cold	Normal	Weak	Cool	Change	22	2
4	98-104	Rainy	Warm	High	Strong	Warm	Same	3	4
5	105-111	Cloudy	Cold	Normal	Weak	Warm	Change	7	0
6	112-121	Rainy	Cold	Normal	Weak	Warm	Same	8	2
7	122-200	Sunny	Warm	High	Strong	Cool	Change	10	69

Definition 3. Let a matrix $M_{i\times j}$ and a matrix $M_{i\times k}$, then their 2-union matrix $M = [M_{i\times j}|M_{i\times k}] = M_{i\times(j+k)}$, where i denotes the row number of the two matrixes, j and k denote the column number of them.

B. FIND-SS algorithm

Considering the first problem, it shares an high-affinity with the noise in real-world datasets. The training examples are very likely to contain some noise or errors in most practical learning tasks. Such inconsistent training examples apparently mislead FIND-S on searching the target hypothesis. The measurement of *support* can be exerted to identify the noisy examples from training datasets. Let n be the number of training examples in T.

$$Support(example\ c_i) = \frac{(X \cup Y) \cdot count}{n} \qquad (1)$$

where the notation X indicates an example as a set of attributes, and the Y is the positive examples; $X.count$ is the number of transactions in T that contain X and $Y.count$ is similar to $X.count$. And the supports of Table 1 are 0.15, 0.175, 0.11, etc. The example set of ID 4 is removed from Table 1 when given a *support* 0.05. Regarding the second problem, in a given training data, the proportion of voting positive can be interpreted as the probability of the positive instance. Hence, example set can be classified easily depending on the percentage of positive examples. Obviously, ID 7 set inclines to a negative classification. Therefore, given a user-specified threshold and a voting scheme, Table 1 can be easily transformed into Table 2.

The measurement of the *support* can also be employed as an input constraint parameter to tune the over-rapid generalization during the mining process. Regarding the third problem, an *Upper Bound* set of hypotheses can be used to supervise the processing of example scan. For example, two constraint hypothesis sets including 1-meta and 2-meta Upper Bound Hypotheses can be given from Table 3(a) and Table 3(b):

The count of 1-meta Upper Bound hypotheses is the total number of all attribute value when each hypothesis only includes one attribute value and some ? (see Table 3(a)) . Similarly, the number of 2-meta Upper Bound hypotheses is C_n^2 where n denotes the count of all attribute values (see Table 3(b)). Each output hypothesis of all steps should be compared with each hypothesis of the specified Upper Bound hypothesis set. However the hypothesis may be converging to one of Upper Bound hypotheses before finishing the task. There is an available solution that is employing a hypothesis set as the finally scan results rather than only one hypothesis. When a hypothesis is generalized to an Upper Bound hypothesis it is added to the set of temporary target hypotheses during learning process. The process is continued recursively until all examples were scanned. And then we devise a similarity-based FIND-S algorithm called FIND-SS (see Algorithm 1).

TABLE 3. UPPER BOUND HYPOTHESES

(a) 1-meta

ID	1-meta hypotheses
1	$< Sunny, ?, ?, ?, ?, ? >$
2	$<?, Warm, ?, ?, ?, ? >$
3	$<?, ?, Normal, ?, ?, ? >$
...	...
m	$<?, ?, ?, ?, ?, Same >$

(b) 2-meta

ID	2-meta hypotheses
1	$< Sunny, Warm, ?, ?, ?, ? >$
2	$< Sunny, ?, ?, High, ?, ? >$
3	$< Sunny, ?, ?, ?, Strong, ? >$
...	...
n	$<?, ?, ?, ?, Cool, Same >$

Algorithm 1 FIND-SS Algorithm

Input:
initialize h to the most specific hypothesis
initialize t_1, t_2 to h as temporary hypothesis
initialize h' to the Upper Bound set
S: Output hypothesis set
1: **for** each two most similar training instances x_i, x_j **do**
2: $t_1 \leftarrow x_i$, $t_2 \leftarrow x_j$
3: $h \leftarrow$FIND-S(x_i, x_j)
4: **if** h is more general than h' **then**
5: add x_i, x_j to S
6: **else**
7: add h to the instance set as a example
8: **end if**
9: **end for**
Remove h in S when $support(h) < threshold$
Output: S

In Algorithm 1, an integer parameter is added as the last element of hypothesis h. The FIND-S(x_i, x_j) denote that the example x_i and x_j apply the FIND-S algorithm for searching the target hypotheses. The $support(h)$ refers to the support count of hypothesis h. To illustrate this algorithm, we assume the training examples are from Table 2 for the EnjoySport task, and the $support$ is given 0.05, then the outputs of each step are as follows:

1) $h \leftarrow < \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing | 0 >$
//initialize h to a most specific hypothesis

TABLE 2. THE NUMBER OF DAYS WITH TRAINING EXAMPLES FOR THE TARGET CONCEPT ENJOYSPORT_NEW.

								Count	
ID	Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	Yes	No
1	1-35	Sunny	Warm	Normal	Strong	Warm	Same	30	5
2	36-73	Sunny	Warm	High	Strong	Warm	Same	35	3
3	74-97	Cloudy	Cold	Normal	Weak	Cool	Change	22	2
5	105-111	Cloudy	Cold	Normal	Weak	Warm	Change	7	0
6	112-121	Rainy	Cold	Normal	Weak	Warm	Same	8	2

$t_1,\ t_2 \leftarrow < \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing | 0 >$
//initialize t_1, t_2 temporary hypothesis
$h' \leftarrow$ 1-meta Upper Bound hypotheses
S : target hypothesis set
2) $t_1 \leftarrow < Sunny, Warm, Normal, Strong, Warm, Same | 30 >$
$t_2 \leftarrow < Sunny, Warm, High, Strong, Warm, Same | 35 >$
3) $h \leftarrow < Sunny, Warm, ?, Strong, Warm, Same | 65 >$
Add h to examples
4) $t_1 \leftarrow < Cloudy, Cold, Normal, Weak, Cool, Change | 22 >$
$t_2 \leftarrow < Cloudy, Cold, Normal, Weak, Warm, Change | 7 >$
5) $h \leftarrow < Cloudy, Cold, Normal, Weak, ?, Change | 29 >$
Add h to examples
6) $t_1 \leftarrow < Cloudy, Cold, Normal, Weak, ?, Change | 29 >$
$t_2 \leftarrow < Rainy, Cold, Normal, Weak, Warm, Same | 8 >$
7) $h \leftarrow <?, Cold, Normal, Weak, ?, ? | 37 >$
Add h to examples
8) $t_1 \leftarrow < Sunny, Warm, ?, Strong, Warm, Same | 65 >$
$t_2 \leftarrow <?, Cold, Normal, Weak, ?, ? | 37 >$
9) $h \leftarrow <?, ?, ?, ?, ?, ?, ? | 112 >$
h is more general than h'
add $t_1,\ t_2$ to S
10) Output $< Sunny, Warm, ?, Strong, Warm, Same | 65 >$,
$<?, Cold, Normal, Weak, ?, ? | 37 >$

The FIND-SS algorithm depicts one way in which the *more_general_than* partial ordering can be used to tune the scan process. The inputs of this algorithm: a *support* threshold, an Upper Bound hypotheses set and a training example set, the output is a series of hypotheses. The algorithm is not only suitable for training the positive examples but the negative ones. In our work, the FIND-SS is implemented to mine the definitional patterns.

C. Example expression and algorithm application

In mining process, the learner is given a set of definitional sentences including broad and narrow ones. These examples are not labeled inside these sentences in any way. And the learner is also not given any seed patterns. Mining definitional patterns from unlabeled sentences and having not any heuristic patterns is very different from the existing research of definition pattern extraction. In other words, the learner only has some definitional samples. The learning task is to discover some high frequency string patterns from given dataset. A pattern it is said to be frequent if it occurs more than a user-specified criterion.

N-grams Segmentation The definitional sentence can be segmented into several strings via the n-grams manner. An n-gram is simply a consecutive sequence of words of a fixed window size n [21]. To depict the working of it, an instance is available that is *The Sun is the star at the center of the Solar System.* which can be represented with twelve 2-gram phrases "The Sun", "Sun is", etc. Note that the example sentence can be segmented into 13-gram to 1-gram including the period. The sequential words also can be seen as a vector. Each definition sentence thus can be formulated as some vectors by given n-grams.

Sentence formalization Each definitional sentence is represented with a series vectors by the n-grams fashion previously. Nevertheless, it is not easy to discover the high-frequency definitional patterns. Document representation model can be utilized to formulate the definitions of the whole dataset. Given a collection of definitional sentences D, let $V = \{s_1, s_2, \cdots, s_{|V|}\}$ be the set of distinctive string in the collection, where s_i is a string. The set V can be called the string of the collection, and $|V|$ is its size, i.e., the number of strings in V. A weight $w_{ij} > 0$ is associated with each string s_i of a definitional sentence $d_j \in D$. If a string appears in definitional sentence d_j, then $w_{ij} = 1$, otherwise $w_{ij} = 0$, i.e.

$$w_{ij} = \begin{cases} 1, & \text{if } s_i \text{ appear in } d_j \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Each definitional sentence d_j is thus represented with a string vector $d_j = (w_{1j}, w_{2j}, \cdots, w_{|V|j})$. Thus, the vectors were grouped and can be represented as a multi-dimensional matrix. An mapping example can be further illustrated from definitional sentences to a matrix. We describe them below.

1) Let (a) to (e) be the definitional sentences.
   a) The Earth is a planet.
   b) Tom will go home. *(noise)*
   c) The Venus is a planet.
   d) The Moon is a satellite.
   e) The dog is a mammal.
2) Vector representation by 2-gram:
   a) $< The\ Earth, Earth\ is, \cdots, planet. >$
   b) $< Tom\ will, will\ go, \cdots, home. >$
   $\cdots$
3) String collection:

$$\begin{aligned} V_2 = \{ &< 0, the\ earth >, < 1, earth\ is >, \\ &< 2, is\ a >, < 3, a\ planet >, \\ &\cdots, \\ &< 17, a\ mammal >, < 18, mammal. > \} \end{aligned} \tag{3}$$

4) Boolean formulation of definitional sentences
   a) $< 1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0 >$
   b) $< 0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0 >$
   $\cdots$
5) Matrix representation

$$M_{n-gram} = M_2 = \begin{Bmatrix} 1 & 1 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 1 & 1 \end{Bmatrix} \tag{4}$$

FIND-SS scan All of the definitional examples were smoothly represented to a multi-dimensional matrix. Each partition string of all definitional sentences is an attribute in V_{2-gram}, and integer symbol 1 and 0 in the matrix can be deemed as the values of corresponding attributes. The pattern learning task thus, is easily transferred to search an acceptable hypothesis using FIND-SS algorithm in matrix. Ginven a given *support* (assumed $support = 0.4$) and 1-meta Upper Bound hypotheses (similar to Table 3(a)), the scan procedures by implementing FIND-SS are as follows:

1) $h \leftarrow < \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing | 0 >$
   $t_1,\ t_2 \leftarrow < \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing, \varnothing | 0 >$
   $h' \leftarrow$ 1-meta Upper Bound hypotheses
   S : target hypothesis set
2) $t_1 \leftarrow < 1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0 | 1 >$
   $t_2 \leftarrow < 0,0,1,1,1,0,0,0,0,1,1,0,0,0,0,0,0,0,0 | 1 >$
3) $h \leftarrow <?,?,1,1,1,0,0,0,0,?,?,0,0,0,0,0,0,0,0 | 2 >$
   Add h to examples
4) $t_1 \leftarrow < 0,0,1,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0 | 1 >$
   $t_2 \leftarrow < 0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1 | 1 >$
5) $h \leftarrow < 0,0,1,0,0,0,0,0,0,0,0,?,?,?,?,?,?,?,? | 2 >$
   Add h to examples
6) $t_1 \leftarrow <?,?,1,1,1,0,0,0,0,?,?,0,0,0,0,0,0,0,0 | 2 >$
   $t_2 \leftarrow < 0,0,1,0,0,0,0,0,0,0,0,?,?,?,?,?,?,?,? | 2 >$
7) $h \leftarrow <?,?,1,?,?,0,0,0,0,?,?,?,?,?,?,?,?,?,? | 4 >$
   Add h to examples
8) Remove $< 0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0 >$
   // $1/5 = 0.2 < 0.4$ (threshold)
9) $S \leftarrow <?,?,1,?,?,0,0,0,0,?,?,?,?,?,?,?,?,?,? | 4 >$

The output vector S is the final output set, where the digital 4 is the count of the examples. For simplicity, we just obtain the index of value 1 in the vector and then find the corresponding attribute name from the V_2, for instance, we can get an attribute, namely ***is a*** from Equation (3). This pattern is in agreement with the fact definitional pattern.

Viewed in toto, unlike FIND-S algorithm, which is severely misled by some noise, FIND-SS algorithm can accommodate some inconsistent data and errors in training dataset. Our results indicate that FIND-SS algorithm employs similarity-based generalization scenario meets the real-world learning purpose.

IV. Experiments

A. Experimental setup

A corpus of 7,935 Chinese sentences, which contains 2,985 definitional and 4,950 non-definitional sentences. The former mostly derived from a random selection of the first sentences of 1791 Wikipedia articles since the first sentences of Wikipedia entries is a definition of the page title in the majority of cases. The articles cover all Chinese 8 categories and 146 subcategories, so as to capture some representative and cross-domain examples of lexical and syntactic patterns. Considering the diversity of patterns, we only derived 1 to 5 sentences per page. The latter, ie. the non-definitional sentences were mainly obtained from the body of Wikipedia articles involving almost all domains. Both definitional and non-definitional sentences, during our learning process, do not require any labeled in POS and inner structure, which is significantly different comparing to the dataset of previous research.

B. Pattern discovering

In this experiment, the dataset of definitions is collected over time, then we can employ the earlier part of the set including 1560 definitional sentences for training, and the later part of the dataset including 1425 definitional sentences for testing. A 10-fold cross-validation method is available and it is performed in the experiment. In this experiment, the maximum length of strings is five, and then each definitional sentence is partitioned separately to some string sequences by 5-gram to 1-gram. The disjunction of all the same length strings consist of a string vector V, and we can get five string vectors. Using Equation (2), all definitional examples are represented to five matrix vectors by different n-grams.

1-meta definitional patterns To discover the 1-meta definitional patterns, the FIND-SS (see Algorithm 1) is implemented to discover the definitional patterns in the vector matrixes. In training process, we let the minimal support be 0.01 based on some experiments, and let the Upper Bound hypotheses is 1-meta. When all rows of the matrix are scanned by the scheme of *the more similar the higher priority* of FIND-SS, the learner gets some vectors including symbol 1, 0 and ?. Some vectors represented by some noisy sentences are removed effectively by the given *minimal support* and *Upper Bound hypotheses*.

The output vectors obtained by FIND-SS can be interpreted to the corresponding strings in collection V according to the index of symbol 1. Not all output strings are suitable as definitional patterns even though parts of them have a high support count in the vector matrix, because some stop words (such as *at, which, on*, etc.), digitals(such as *1, 2, one, 4th*, etc.) and name entities (such as *the names of persons, organizations, locations, expressions of times, quantities*, etc.) have a poor classification behavior between the definitional and the non-definitional sentences. Hence, some stop words, common punctuation and marks and numerical symbols would be filtered out from the preliminary patterns set. Therefore, only small part of patterns are selected.

It is based on the assumption of the more complex patterns, the lower recall and the higher precision. So there is an available method to obtain the patterns that are different in length metric to discover an optimization pattern set. Parts of the definitional patterns can be seen in Table 4, where the strings of first column are Chinese definitional patterns, the second column strings are the corresponding English patterns and the last column strings are the examples.

2-meta definitional patterns Basically, the set of 1-meta definitional patterns can take effect for recalling the candidate definitions in corpora. For further optimizing the pattern set and reducing the percentage of noisy sentences, some 2-meta definitional patterns (see Table 5) can be also learned by FIND-SS algorithm based on the given support (also 0.01) and 2-meta Upper Bound hypotheses.

Considering the training matrix, there is a small difference between the learning process of 1-meta and 2-meta. The former searches the target vectors in the matrix that is represented by one of n-grams. Nevertheless the latter searches them from the called 2-union matrix (see Definition 2) that is combined by any two different n-gram matrixes. Therefore, n different n-gram matrixes can be combined C_n^2 2-union

TABLE 4. PARTS OF 1-META PATTERNS FROM TRAINING DATASETS.

Chinese patterns	Corresponding English	English definitional sentences
),是一种	), is a/an	Germany(Deutschland**), is a** European country.
),是一	), is a/an	Germany(Deutschland**), is a** European country.
),又称	), also called	Massive open online courses (popular in USA**), also called** MOOCS.
)是一个	) is a/an	Germany(Deutschland) **is a** European country.
)是一种	) is a/an	Germany(Deutschland) **is a** European country.
,指的是	, refers to	Pelvic pain**, refers to** pain in the abdomen below the belly button.
)是指	) means	A laundry room (utility room) **means** a room where clothes are washed.
...	...	...
是	is	The Sun **is** the star at the center of the Solar System.
为	termed	Self-hypnosis is also **termed** autogenous training.
指	denoted	A room **denoted** any distinguishable space within a structure.
称	called	Each group of memory cells is **called** as a node.
:	:	Germany**:** a European country.
系	is	Stock market **is** a market of stock.
即	i.e.	Stock market **i.e.** a market of stock.

TABLE 5. PARTS OF 2-META PATTERNS FROM TRAINING DATASETS.

Chinese patterns	Corresponding English	English definitional sentences
是···研究	is···study	Economics is that it **is** the **study** of the economy.
是···的学科	is···discipline	Communications **is** a brand-new **discipline**.
是···的科学	is···science	Philosophy **is** a nomothetic **science**.
是···的方法	is···method	Vitreoretinal surgery **is** an effective **method** to arn.
是···的过程	is···process	Biological evolution **is** one evolutionary **process**.
是···的行为	is···behavior	Cheating **is** considered as an unforgivable **behavior**.
为···是一种	is···as a	Depression **is** widely recognized **as a** disease.
系···研究	is···study	Genetics **is** the **study** of the general laws of genetic.
称···包括	is called ···include	The traditional American family **is called** nuclear family, which **includes** the ···
指···研究	refer to ···study	Genetics **refers to** the **study** of the general laws of genetic.
是指···又称	refer to ···also called	Personality barrier **refers to** the ill ··· it is **also called** personality ···
研究···的学科	study ···subjects	Modern CNC technology is the **study** of a high-precision motion control **subjects**.
研究···的科学	study ···science	Psychology is a **study** of human mental activity of the laws of **science**.
...	...	...

matrixes. In our experiment, it will obtain ten 2-union matrixes for training from 1-gram to 5-gram original matrixes, such as $[M_{1-gram} \mid M_{2-gram}]$, $[M_{1-gram} \mid M_{3-gram}]$, $\cdots$, $[M_{4-gram} \mid M_{5-gram}]$. Besides the scan matrixes, the attribute vectors ie. string sequences also have a small difference in 1-meta and 2-meta pattern mining. We draw a table of scan matrixes and attribute vectors to illustrate their differences (see Table 6).

C. Comparative Evaluation

Experiments were performed with 10-fold cross validation, we obtained 135 definitional patterns including 97 1-meta and 38 2-meta patterns, which is more than the sum of all previous Chinese definitional patterns (less than 50). To assess the capability of FIND-SS, we calculated respectively the measures of recall, precision and F-score using our and previous patterns in the test dataset, in which consist of 1425 definitional and 4950 non-definitional sentences. We also performed a comparison with 3 research teams, a state-of-the-art definition extraction.

As mentioned earlier, candidate definitions were discovered from textual data by the given patterns. A remarkable definitional patterns should be able to cover as many definitions as possible (recall) and as few non-definitions as possible (precision). Given the same *support*, the 1-meta pattern set cover more definitions than the 2-meta set because the former patterns is more general than latter's. In test phase, the 1-meta definitional pattern set are tested regarding the key purpose of candidate definition extraction.

To compare the performances, we report the the state-of-the-art definition extraction from Table 7(a) to 7(c). The digital in *Length* column (column 1) of the table denotes the different subsets, which are grouped by one or several sizes. For example, the "3" is a subset that consist of such patterns of a fixed window size three, while the "3-5" refers to the patterns of size three to five. The *Positive* column says the count of positive definitions, while the *Negative* is the count of non-definitions via the corresponding pattern set. The *P*, *R* and *F* are represented respectively by *Recall*, *Precision* and *F-score*. The *avg.* row lists respectively the average values of *Recall* and *Precision* and the last column of it is the F-score calculated via the *Recall* and *Precision* in its same row.

Fangyie Leu's Patterns: FangYie Leu et al. [16] proposed a system to extract term definitions from the Web based on the given Chinese terms. They used 19 Chinese definitional patterns including common and domain ones for analyzing the their behavior. It is noteworthy that these pattern of them are collected by manual labour from newspapers and magazines. In Table 7(a), only "2" set gets an acceptable performance, namely 89.26% precision, while the recall is only 15.16%.

Endong Xun's Patterns: Endong Xun et al. [22]–[27] used 40 definitional patterns all together for extracting definitions. However, as with Jinfeng Tian's method, these patterns are also collected by manual extraction in scientific and technological fields. In Table 7(b), the "4" and "4-5" pattern sets obtain 100% precision, while the corresponding recalls of them are only 0.84%.

TABLE 6. THE COMPARISON OF 1-META AND 2-META IN VECTOR MATRIXES AND ATTRIBUTE VECTORS.

Type	1-meta	2-meta
Vector Matrix	M_{i-gram}, where $1 \leq i \leq n$	$[M_{i-gram} \mid M_{j-gram}]$, where $1 \leq i < j \leq n$; M_{i-gram} or M_{j-gram}, where $1 \leq i = j \leq n$
Attribute Vector	V_{i-gram}, where $1 \leq i \leq n$	$V_{i-gram} \cup V_{j-gram}$, where $1 \leq i \leq j \leq n$

TABLE 7. PERFORMANCE COMPARISON

(a) Performance of Fangyie Leu's Patterns

Length	Positive	Negative	R (%)	P (%)	F (%)
5	/	/	/	/	/
4	3	2	0.21	60.00	0.42
3	12	7	0.84	63.16	1.66
2	216	26	**15.16**	**89.26**	25.91
1	1080	2011	75.79	34.94	47.83
4-5	3	2	0.21	60.00	0.42
3-5	15	8	1.05	65.22	2.07
2-5	229	32	16.07	87.74	27.16
1-5	1088	2025	76.35	34.95	47.95
Avg.			23.21	61.91	33.76

(b) Performance of Endong Xun's Patterns

Length	Positive	Negative	R (%)	P (%)	F (%)
5	0	0	0.00	/	/
4	12	0	**0.84**	**100.00**	1.67
3	103	38	7.23	73.05	13.15
2	560	257	39.30	69.54	49.96
1	1144	1349	80.28	45.89	58.40
4-5	12	0	**0.84**	**100.00**	1.67
3-5	103	38	7.23	73.05	13.15
2-5	593	281	41.61	67.85	51.59
1-5	1158	1492	81.26	43.70	56.83
Avg.			28.73	71.51	40.99

(c) Performance of Jinfeng Tian's Patterns

Length	Positive	Negative	R (%)	P (%)	F (%)
5	/	/	/	/	/
4	13	0	**0.91**	**100.00**	1.81
3	111	19	7.79	85.38	14.28
2	605	257	42.46	70.19	52.91
1	1168	1597	81.96	42.24	55.75
4-5	13	0	**0.91**	**100.00**	1.81
3-5	111	19	7.79	85.38	14.28
2-5	629	261	44.14	70.67	54.34
1-5	1181	1701	82.88	40.98	54.84
Avg.			33.61	74.36	46.29

(d) Performance of Our Patterns

Length	Positive	Negative	R (%)	P (%)	F (%)
5	2	0	0.14	100.00	0.28
4	98	0	6.88	100.00	12.87
3	535	263	37.54	67.04	48.13
2	1028	763	**72.14**	57.24	**63.83**
1	1346	2519	94.46	34.83	50.89
4-5	98	0	6.88	100.00	12.87
3-5	535	263	37.54	67.04	**48.13**
2-5	1061	860	74.46	55.23	63.42
1-5	1376	2770	**96.56**	33.19	49.40
Avg.			**47.40**	68.29	**55.96**

Jinfeng Tian's Patterns: Comparison with Fangyie Leu and Endong Xun, Jinfeng Tian et al. [28] further developed the pattern collection on the basis of predecessor. A few patterns from history and geography fields are expended to the pattern set. Nevertheless, we note that the means of pattern acquisition of the them share the common method, ie. manually identifying definitional patterns from textual text. The "4" and "4-5" pattern sets obtain 100% precision, while the corresponding recalls of them are also very low, only 0.91% (see Table 7(c)).

Our Patterns: In Table 7(d), the "2" pattern set performs best, obtaining 72.14% precision, 57.24% recall and 63.83% F-score. The "1-5" pattern set obtains 96.56% recall, in addition, the "4", "5" and "4-5" sets get 100% precision. Comparing with the three research teams, we acquire a much higher recall, namely above 96% on "1-5" pattern set, while the best performance of the former three is only 82.88%. We also notice that our precision of "1-5" set is lower than previous' since our set contains more patterns. However, it accommodates the low precision based on the common consensus that is better high recall and low precision than high precision and low recall in candidate definition extraction. Regarding the longer patters, especially the "3-5" set, we can see that our F-score (48.13%) is at least 43 percent higher than any other behavior (2.07%, 13.15% and 14.28%). Taking into account the average measures of R, R and F, our have a slightly lower precision, but both recall (47.40%) and F-score (55.96%) are significantly higher than previous performances.

V. CONCLUSIONS AND DISCUSSION

In this paper, we have presented a similarity-based FIND-S algorithm, named FIND-SS to automatically extract definitional patterns. The **novelty** of our approach is:

Similarity Priority: The scenario of *the more similar the higher priority* is performed in FIND-SS algorithm. The approach always selects two most similarity hypotheses for generalization, while FIND-S traverses the example hypothesis only according to priority of their appearance in the training dataset. The latter has to terminate its trace before finishing the process if the training dataset contains some inconsistent examples. By contrast, the former can search all examples even though the training data is noisy. In addition, it can output effectively a series of hypotheses, which is in agreement with the real-word data.

Support and Vote: In a learning purpose, a pattern covering too few cases may not be useful because it may just occur due to chance. The vote offers a feasible measure to output the majority vote.

Upper Bound Hypotheses: The upper bound hypotheses are acceptable and limited general hypotheses, which include 1-meta hypotheses, 2-meta hypotheses, etc. Because the learning object of FIND-S is only one hypothesis, the final hypothesis is an over generalization hypothesis if the training examples contain some negative examples. We place special emphasis on the learning purpose of FIND-SS is a set of hypotheses rather than only a hypothesis. These Upper Bound hypotheses is performed to constrain the scan of FIND-SS to tune the target hypotheses, which is available for real-world dataset.

*Unlabeled:*The approach is independent of any annotated data not only the label of lexico-syntactic structure but the label of definiendum, definiens and connector within definitional sentences. Furthermore, it does not require any pattern seeds for pre-processing. Only given the positive definitions and a dictionary of stop words in NLP, it can automatically learn the definitional patterns.

High Performance: The ability of the FIND-SS is to mine the definitional patterns in cross-domain. 97 1-meta and 38 2-meta common patterns are learned, which is about triple volumes of previous Chinese definitional patterns. The patterns, especially the longer's have a high performance as compared with all previous Chinese patterns.

Taken together with our and previous approaches, our work is the first report that is automatic learning Chinese definitional patterns. Furthermore, our experimental results show that our patterns have a significantly higher performance than the previous ones. Nevertheless, there is also a **limitation** we should point out and try to introduce a solution for it.

Combinatorial Explosion: For learning more 2-meta features, both high and low frequent (the minimal support is very low) patterns are considered, provided that they can recall more definitional sentences. Nevertheless, it may cause combinatorial explosion and make extracting impossible because the number of patterns will grow linearly with the total size of the training examples. An available solution to this problem is to partition the training examples into several smaller subsets (blocks) by randomly, each of which is mined separately via the algorithms. And then some preliminary patterns are generated by disjunction of the mining results of all subsets.

In order to support future efforts we are releasing our datasets including all pattern sets as a freely available resource from http://cit.sjtu.edu.cn/defi_patterns.rar. In near future, we will further extend this approach to learn the non-definitional patterns, and merge them into the training phase for accessing and optimizing the common definitional patterns, accordingly, to improve the performance of definition extraction.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (NSFC) under the Grant No. 61375053.

REFERENCES

[1] F. De Benedictis, S. Faralli, R. Navigli *et al.*, "Glossboot: Bootstrapping multilingual domain glossaries from the web," in *Proceedings of the 51st Annual Meeting of the ACL*, 2013, pp. 528–538.

[2] E. Westerhout, "Definition extraction for glossary creation: A study on extracting definitions for semi-automatic glossary creation in dutch," *Lot Dissertation Series*, vol. 252, pp. 13–27, 2010.

[3] C. Borg, "Automatic definition extraction using evolutionary algorithms," Ph.D. dissertation, University of Malta, 2009.

[4] O. Trigui, L. H. Belguith, and P. Rosso, "An automatic definition extraction in arabic language," in *Natural Language Processing and Information Systems*. Springer, 2010, pp. 240–247.

[5] J. Zhang, Y. Wang, and H. Wei, "An interaction framework of service-oriented ontology learning," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 2303–2306.

[6] A. Jimeno-Yepes, R. Berlanga-Llavori, and D. Rebholz-Schuhmann, "Ontology refinement for improved information retrieval," *Information Processing & Management*, vol. 46, no. 4, pp. 426–435, 2010.

[7] R. Gil and M. J. Martín-Bautista, "A novel integrated knowledge support system based on ontology learning: Model specification and a case study," *Knowledge-Based Systems*, vol. 36, pp. 340–352, 2012.

[8] S. Faralli and R. Navigli, "A new minimally-supervised framework for domain word sense disambiguation," in *Proceedings of the 2012 Joint Conference on EMNLP and Computational Natural Language Learning*. ACL, 2012, pp. 1411–1422.

[9] P. Velardi, S. Faralli, and R. Navigli, "Ontolearn reloaded: A graph-based algorithm for taxonomy induction," *Computational Linguistics*, vol. 39, no. 3, pp. 665–707, 2013.

[10] T. Flati and R. Navigli, "Spred: Large-scale harvesting of semantic predicates," in *Proceedings of 51st Annual Meeting of the ACL*. ACL, 2013, pp. 1222–1232.

[11] S. Gerardo, P. Mara, and T. Juan-Manuel, "Foreword of international workshop on definition extraction," in *Workshop On Definition Extraction 2009*, 2009, pp. i–iii.

[12] R. Navigli and P. Velardi, "Learning word-class lattices for definition and hypernym extraction," in *Proceedings of the 48th Annual Meeting of the ACL*. ACL, 2010, pp. 1318–1327.

[13] B. Liu, C. W. Chin, and H. T. Ng, "Mining topic-specific concepts and definitions on the web," in *Proceedings of the 12th international conference on WWW*. ACM, 2003, pp. 251–260.

[14] A. Przepiórkowski, Ł. Degórski, B. Wójtowicz, M. Spousta, V. Kuboň, K. Simov, P. Osenova, and L. Lemnitzer, "Towards the automatic extraction of definitions in slavic," in *Proceedings of the Workshop on Balto-Slavonic NLP: Information Extraction and Enabling Technologies*. ACL, 2007, pp. 43–50.

[15] A. Przepiórkowski, Ł. Degórski, and B. Wójtowicz, "On the evaluation of polish definition extraction grammars," in *Proceedings of the 3rd Language & Technology Conference*. Poznan, Poland, 2007, pp. 473–477.

[16] F.-Y. Leu and C.-C. Ko, "An automated term definition extraction system using the web corpus in the chinese language," *Journal of Information Science & Engineering*, vol. 26, no. 2, pp. 505–525, 2010.

[17] C. Borg, "Discovering grammar rules for automatic extraction of definitions," *Doctoral Consortium at the Eurolan Summer School*, pp. 61–68, 2007.

[18] C. Borg, M. Rosner, and G. J. Pace, "Automatic grammar rule extraction and ranking for definitions," in *LREC*, 2010, pp. 2577–2584.

[19] S. Faralli and R. Navigli, "A java framework for multilingual definition and hypernym extraction," in *Proceedings of the 51st Annual Meeting of the ACL, System Demonstrations*. Citeseer, 2013, pp. 103–108.

[20] T. M. Mitchell, "Machine learning. 1997," *Burr Ridge, IL: McGraw Hill*, vol. 45, 1997.

[21] B. Liu, *Web data mining: Exploring Hyperlinks, Contents and Usage Data*, 2nd ed. Springer, 2011.

[22] L. C. Xun Endong, "Applying terminology definition pattern and multiple features to identify technical new term and its definition," *Journal of Computer Research and Development*, vol. 46, no. 1, pp. 62–69, 2009.

[23] J. A. R. S. Xu Yong, Xun Endong, "A web-based term definition extracting system," *Journal of Chinese Information Processing*, vol. 18, no. 4, pp. 37–43, 2004.

[24] S. R. Zhang Rong, "Internet based chinese term definition," in *The 8th Chinese Conference on Computational Linguistics*, 2005, pp. 428–434.

[25] J. Aiping, "The pattern research of term definition in the scientific literature," Master's thesis, Beijing Language and Culture University, 2002.

[26] Z. Rong, "Research on extraction and clustering of term definition and term extraction," Ph.D. dissertation, Beijing Language and Culture University, 2003.

[27] S. R. Zhang Rong, "Research on extraction of term definition," *Terminology Standardization and Information Technology*, vol. 2006, no. 1, pp. 29–32, 2006.

[28] H. H. L. W. Tian Jinfeng, Zeng Xihong, "Research on automatic construction of definition notes for concepts in ontology thesaurus," *Library and Information Technologys*, vol. 2011, no. 11, pp. 9–16, 2011.