Identifying Disease Definitions with a Correlation Kernel for Symptom Extractions from Text

Minsu Ko*, Sung-Hyon Myaeng†
Division of Web Science and Technology* & Department of Computer Science†
Korea Advanced Institute of Science and Technology
Daejeon, Republic of Korea
Email: {ryan0802, myaeng}@kaist.ac.kr

Abstract—Since most health-related knowledge is created by experts, it is not easy for general public to access, understand, and utilize such knowledge in daily living. It would be most convenient and useful to a healthcare knowledge base that a user can easily start exploring from symptoms and arrive at candidate diseases and eventually obtain knowledge for treatment and prevention. We have embarked on a project whose goal is to build such a healthcare knowledge base from text by using natural language processing and text mining techniques. This paper focuses on how definition sentences can be detected and describes a method of ranking sentences based on the degree to which they contain definitions of diseases, which should contain symptom information. While our work is basically to build a classifier that identifies definition sentences, the main contribution lies in the development of a new kernel method that utilizes correlations among different types of tokens. We evaluated our method to arrive at a conclusion that the proposed method can be very effective with a training data that is almost an order of magnitude smaller than the method of using dependency parser.

Index Terms—symptom extraction; definition sentence; collocation; colligation; correlation kernel;

I. INTRODUCTION

Effective dissemination of healthcare information to the public is essential not only for individuals’ well-being but also for national efforts to reduce health-related cost. Wide spread of healthcare information through the Web and other media provides a great potential that people can access health-related information at a level never experienced before. Despite the affluence of information, however, availability does not necessarily mean that the information is in the form that can be accessed and consumed easily for practical purposes. This is because 1) health-related information is usually created by experts and hence not easily comprehensible by laypeople without concentrated efforts, 2) various aspects of disease information are dispersed in various media and therefore need to be integrated, and 3) symptoms are the starting point for people to search, gather, and learn about the related disease, possible treatments, potential health risks, etc. While symptom descriptions are the entry point to the vast health-related information, they are usually embedded in definitions and/or descriptions under a disease name.

As an attempt to make healthcare information more readily available and usable, we have a long-term research goal of building an explorable knowledge base and a question-answer system. By organizing healthcare knowledge that is extracted and integrated automatically from free text including web documents, we aim at providing a user with the capability of exploring the knowledge base from symptoms and arrive at candidate diseases and eventually obtain knowledge about treatment, prevention, therapies, etc. This capability is to be integrated with a healthcare question-answering system to maximize the ease of utilizing such knowledge.

Having observed that disease definition sentences frequently contain symptom descriptions in terms of cognitive/perceptive/reactive/temptation aspects together with other essential elements such as the malfunctioning organs or mechanisms, this paper describes our effort in identifying such definition sentences in the vast amount of unstructured health-related text. While identifying definition sentences from an encyclopedia such as Wikipedia seems relatively straightforward because they usually appear in the first sentence, it turns out that the task includes computationally challenging issues when we have to handle general web documents. As in Fig. 1, the locations of the definition sentences are quite dispersed. Given that the red dotted line cuts the data into two halves, we can observe that a half of the sentences are found below the first top one third of the locations. The challenges also stem from the variability of natural language utterances. Because of the differences in writing styles, definition sentences have highly different syntactic structures, unlike those found in dictionaries [15]. Definition sentence may explain a symptom using entities including the keyword’s hypernym and phrases describing the function [12], and the authors have a complete freedom of using either technical terms or plain expressions.

In this paper, we structure the problem as a classification task between definition sentences (DEF) and elaboration sentence (ELA). It would be interesting to analyze the linguistic characteristics of definition sentences for which authors attempt to be careful in selecting the right words for the genus and differentia in defining the disease and its symptoms, our current focus is to build a machine learning-based classifier. Instead of blindly using words as features, however, we attempt to use lexico-syntactic features that constitute the characteristics of definition sentences. We propose an weighted correlation kernel for SVM classification. Another unique aspect of our research is to make the resulting classifier work
for multiple languages having different syntactic structures.

II. PROPOSED APPROACH

A logical approach to the problem of learning the characteristics of definition sentences for supervised machine learning classifier would be to capture their structural patterns to build a model. An addition to this would be to consider extracting paraphrases for the sentences in the training data. Using structural patterns has been materialized as a kernel in SVM classifiers for similar tasks. For instance, a parse-tree kernel was successfully introduced for PPI extraction [1] by formulating it as a classification problem. However, a past study showed that syntactic structures and their sequences alone were not effective especially when both unlexicalized and lexicalized subtree kernels were used [21]. For more general setting, it was shown that a parse-tree kernel was no better than a built-in kernel [2] although the experiment was conducted in a somewhat limited way.

A. Combining syntactic and lexical co-occurrence features

As a way to examine the possibility of using parse-tree kernel for our classification problem, we analyzed our dataset to see if the syntactic patterns between definition and elaboration sentences are distinct. As in Figure 2, however, the distribution patterns of the two types of sentences over different syntactic constituents were quite similar to each other. The X-axis and Y-axis represent different syntactic chunks in the order of occurrence frequency and the density or frequency of definition sentences, respectively. While the connecting dots represent elaboration sentences, the red line is the result of black dots representing definition sentences. This analysis led us to believe that syntactic patterns alone would not be good discriminating features for our classification task and that we need to devise a method of using additional features in conjunction with a parse-tree kernel in order to make the two types of sentences more distinct.

Another problem we encountered in using a supervised classifier like SVM is the fact that it would be difficult to acquire a large quantity of training data. Because of the nature of definition sentences, they occur only once or a few times in a document describing a disease. This implies that our feature extraction method must be different from those of the others where many training instances can be acquired for a syntactic kernel. It is necessary to make use of additional information such as term co-occurrence information. Our decision to use such additional information is in addition to our question whether the syntactic structure is sufficient in distinguishing definition sentences from the rest.

B. Linguistic clues for different information types

In our task of identifying definition sentences for diseases and health problems, we assume that the names of diseases and problems (keywords hereafter) must exist in text for which definitions are sought. As such, we consider the co-occurrence information between a keyword and other words in the training instances and use the related distributions as features for classification. This is because even when a keyword is found in a sentence, it is not sufficient to make a conclusion; it’s necessary to know the type of information being conveyed in the sentence. In other words, we need to differentiate between definition sentences and elaboration sentences even after collecting all the candidates that contain a keyword and/or some syntactic structure that exist in documents of the healthcare domain.

Since a definition sentence should contain new information, it has an appropriate article in case of English and most European languages or starts with a keyword followed by an appropriate case marker in agglutinative languages. Since an elaboration sentence, on the other hand, builds on top of the definition sentence, it usually contains a definite article or a determiner in the former language family or a different case marker in the latter. Considering these linguistic phenomena, we propose a correlation kernel that rests on statistical correlation between a keyword and other words in distinguishing definition sentences from the rest.

C. Capturing co-composition relationships

Co-composition is a concept that posits two constituents in a sentence may have bidirectional function application relationships unlike a usual relationship where one plays a
role of a function and the other an argument in compositional semantics. Since both constituents play a function role, their inter-dependency is stronger than usual [18]. For example, in a head-complement dependency, not only the head imposes constraints on the complement, but also the complement imposes linguistic requirements on the head [6]. From the perspective of subcategorization, a word subcategorizes for both its head and complement. In our context, the classifier needs to take into account and learn such co-composition relationships. We do that by considering both content words and function words in characterizing the co-occurrence patterns associated with a keyword. In other word, collocations and colligations are used in defining the kernel that help characterizing the definition sentences for the keyword.

D. Extracting definition sentences

For the purpose of identifying and utilizing linguistic clues for the purpose of distinguishing definition and elaboration sentences, we extract patterns of case markers, or Josa in Korean, as in a previous study where the goal was to identify paraphrases of definition sentences in Japanese [4]. Note that both Korean and Japanese are agglutinative languages and have similar grammar structures. Table I contrasts definition and elaboration sentences both in Korean and their translations into English.

We built a classifier for the purpose of automatically extracting definition sentences, by proposing a supervised learning method that learns lexical semantic contextual information. More precisely, we propose a correlation kernel that can be computed efficiently. The kernel was envisaged with the help of a previous study [23] on extracting term definitions. It attempted to extract broadly defined Korean definition sentences based on four morpho-syntactic patterns as in Table II, which help characterizing diverse possible types of definition sentences. However, we felt that they were not sufficient to automatically detect such sentences and hence developed a machine leaning approach where the patterns help developing the features.

Table I

<table>
<thead>
<tr>
<th>Type</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Definition</td>
<td>Osteoporosis is a disease characterized by the reduced bone density and solidity resulting in increased possibility of bone fracture.</td>
</tr>
<tr>
<td>Elaboration</td>
<td>Patients with osteoporosis face fracture risk with no presymptoms, and such fracture is followed by a sharp pain and a longer treatment period.</td>
</tr>
</tbody>
</table>

Table II

<table>
<thead>
<tr>
<th>Type</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Term -[TM] [Rel N] -[Copula]+FE</td>
</tr>
<tr>
<td>B1</td>
<td>[[Rel N] -[Copula]+RE Term] -[TM]</td>
</tr>
<tr>
<td>C1</td>
<td>Term -[TM] VP -E</td>
</tr>
<tr>
<td>C2</td>
<td>Term -[TM] NP -[Copula]+FE</td>
</tr>
<tr>
<td>D1</td>
<td>NP -[Copula]+RE Term</td>
</tr>
</tbody>
</table>

III. CORRELATION KERNEL

The proposed kernel can be considered as an extension of the convolution parse-tree kernel [3] where the main idea is to sever a parse-tree into its sub-trees and transfer it as a point in a vector space in which each axis denotes a particular sub-tree in the entire set of parse trees. If this set contains $M$ unique sub-trees, the vector space becomes $M$-dimensional. The similarity between two parse trees can be obtained by computing the inner product of the two corresponding vectors, which is the output of the parse-tree kernel. In other words, the original data is mapped into feature space via mapping $\phi$ that is assessed via the kernel function, and classification decision is made in the feature space.

The main thrust of the proposed kernel lies in the following patterns.

- Syntactic patterns around a keyword, which are to be distinguished from the common method of using word-based features including $n$-grams
- Co-occurrence between a keyword and the linguistic clues such as articles and case markers that differentiate definition from elaboration sentences
- Co-occurrences between a keyword and other words of two types: content words and non-content words

To make a distinction between two types of co-occurrences, the former (with a content word) and the latter (with a non-content word) are referred to as collocation (COL) and colligation (CLG), respectively. The definition sentence is weighted according to the degree of correlation by two compositional
categories because the input data of text documents cannot readily be described by explicit feature vectors.

For the syntactic patterns, once a keyword is identified in a definition sentence used as a training instance, we examine if the sentence starts with the keyword having the syntactic tag of NP-SBJ (noun phrase taking the syntactic role of subject) and generate a derivational sequence of supertags of the remaining words of the sentence. In this case, we use the supertags before and after the keyword. Sequences are weighted according to the degree of correlation by two compositional categories. The idea is that the more subsequences two sentences have in common, the more similar they are considered in matching.

There are three issues that need to be resolved in designing a kernel. The first issue is related to normalization of $\chi^2$-distributions. The second one is to handle the divergence between training and testing data. Because the distribution of features in the training data is denser with a large amount, it would be difficult to generate from the test data a vector representation of the same density. Therefore, it is necessary to measure the distributional differences between them and transform the test data distribution by applying a decay factor reflecting the divergence. This process has an effect of handling relatively sparser test data.

Finally, the third issue has to do with probabilistic regularized confidence. Although it is desirable to produce calibrated posterior probability values as a result of a SVM classifier, they are usually not available in standard SVM classifiers [17]. For the problem of definition sentence identification, we prefer degrees of belief for individual output rather than a binary decision. We adopt the solution suggested by [17], which integrates SVM and Sigmoid function.

Following is a detailed description of how our correlation kernel was constructed. Given a collection of definition sentences in the training data, our task is to generate new features that are based on correlation of token pairs of different kinds.

For a sentence containing a keyword $h$, let us define two vectors, $C$ and $G$, of collocation tokens and colligation tokens, respectively, as follows:

$$C = [c_1, c_2, \ldots, c_i]^T, \quad G = [g_1, g_2, \ldots, g_j]^T$$  (1)

where $i$ and $j$ represents the number of tokens excluding the keyword in the sentence. An element is 1 or 0 depending on whether it is collocation or colligation in $C$ and vice versa in $G$. Features are generated by forming all possible pairs $(x_1, x_2)$ of three kinds:

1) $x_1 = h \& x_2 = c_i \in C$ for all $i = 1, \ldots, n$
2) $x_1 = h \& x_2 = g_j \in G$ for all $j = 1, \ldots, n$
3) $x_1 = c_i \& x_2 = g_j \in C$ for all $i, j = 1, \ldots, n$ and $i \neq j$

For correlation between each pair of tokens as defined above, $corr(x_1, x_2)$ is computed using Pearson’s $\chi^2$, a statistical test commonly used to compare observed data with data we would expect to obtain according to a specific hypothesis. $\chi^2$ is the sum of the squared difference between observed and the data, divided by the expected data in all possible categories. [19]

$$\chi^2_{h, cor} = \frac{N((O_{11}O_{22} - O_{12}O_{21}) - N/2)^2}{R_1R_2C_1C_2}$$  (2)

where $O_{11}$ and $O_{22}$ represent the cases of two items occurring together and not occurring at all, respectively, and $O_{12}$ and $O_{21}$ represent cases of only one of them occurring in the data, respectively. In a contingency table, $R$ and $C$ in the denominator represent row and column, and the numbers the index values of the contingency table.

To compute $\chi^2$ between two, we counted the occurrences of $X_1$ and $X_2$ from the training corpus by setting a sliding window of 13 tokens.

With the center item $p$ as the pivot, we scan through the left to find $X_1$ and the right to find $X_2$. The first window moves to the right by one token ignoring sentence boundaries until it reaches the end of the corpus.

There are three issues that need to be resolved in designing the kernel. The first issue is related to normalization and transformation of $\chi^2$ distributions to make the values usable for SVM training. We normalize the $\chi^2$ values so that the results have a range $[0,1]$ as follows:

$$\chi^2_{norm[0,1]} = \frac{\chi^2 - \min(|\chi^2|^T)}{\max(|\chi^2|^T) - \min(|\chi^2|^T)}$$  (3)

As a result, we obtain

$$corr(X_1, X_2) = \chi^2_{norm}(X_1, X_2)$$  (4)

One problem with the result using normalized $\chi^2$ is that a significant portion of the correlation values are close to 0.0, making the features meaningless for SVM learning. For a remedy, we transform the correlation data to generate the final feature values as in [20].

$$\lambda(X_1, X_2) = e^{-\gamma(1-corr(X_1, X_2))}$$  (5)

The difference is shown in Figure 3 where the curves at the bottom are the normalized correlation values whereas those in the middle are the result of transformation. The second issue is to handle the divergence between training and testing data. Because the distribution of features in the training data is denser with a large amount, it would be difficult to generate from the test data a vector representation of the same density. Therefore, it is necessary to measure the distributional differences between them and transform the test data distribution by applying a decay factor reflecting the divergence. This process has an effect of handling relatively sparser test data.

IV. EXPERIMENTS

We evaluated the proposed method for the definition sentence identification task by comparing it against the built-
in kernels in the SVM package we used, n-gram kernels relying on token sequences [7], and the syntax-tree kernels. We crawled text data from two largest portals in Korea \(^2\), for 20 diseases. Table III shows the numbers of documents (#Docs) and sentences containing definition (#D) and elaboration (#E) for each disease. A total of 3,000 sentences form our training and testing data, where a half contains definition sentences and the other elaboration sentences for each disease.

The first experiment focused on the performance of SVM’s two built-in kernels for making a bottom line and then we checked the performance of the n-gram method for generating rich sequential features as a baseline. The third experiment focused on the performance of the parse tree kernels using its subset trees and finally we checked the performance of our proposed method.

A. Dataset

We constructed a corpus consisting of both definition and elaboration sentences for 20 diseases, which were extracted from online forum pages created between January, 2008, and March, 2013, in two largest portals in Korea. Sentences were first automatically selected based on some patterns and manually examined to produce the final gold standard set for definition sentences. For elaboration sentences, the next three sentences from each definition sentence were selected as candidates, which were examined manually to produce the final set.

More specifically, we crawled all the forum pages that contain a disease keyword and selected candidate sentences by checking existence of a keyword having a syntactic tag NP-SBJ (noun phrase and subject of a sentence) that reveal definition sentences. Given that the Korean language uses case markers heavily, we relied on the existence of subjective case markers, ‘-’(-un) (-un)’, ‘-’(-nu)’, ‘-’(-o)’ and ‘-’(-il)’ that indicate contrastive focus. In European languages, the case markers are manifested as indefinite articles. Other subjective markers, ‘-’(-i)’ and ‘-’(-ka)’, were excluded from the patterns because they are used to provide additional information for nouns that have been introduced in the discourse. They correspond to the definite articles in European languages.

In order to construct a collection for both training and testing, we collected a set of candidates for both definition and exploration sentences by using the patterns in Table II. They were given to two human judges for determining whether the sentences belong to definition or elaboration classes. The inter-judge agreement was measured in terms of F-measure \(^3\), instead of Cohen’s Kappa [24] that is often used, because that latter is known to be inappropriate for a dataset with some skewed distribution across true and false values ([25], [26]). For definition sentences in our dataset, for example, the proportion of the true definition sentences is much larger than that of false ones and the judges tend to agree to each other. The average F1-scores for definition and elaboration sentences across 20 diseases are respectively 0.945 and 0.848. F-measure is calculated as follows:

\[
F1 = \frac{2 \times tp}{pp + rp} \tag{6}
\]

B. Experimental Set-up

For parsing sentences, we employed a Korean dependency parser we developed. For kernel-based learning, we used SVM-light 6.02 \(^4\) and added the parse-tree kernel based on [1] and the proposed correlation kernel. For evaluation of classification performance or accuracy, we used the average loss value provided by the package. We followed the same regularization parameter (\(C\)-value) set to 300 as in [1]. All the experiments were run with 10-fold cross validation. The collection of 3,000 sentences consisting of both definition and elaboration sentences were divided into 10 parts, each of which has the same number of definition and elaboration sentences. Nine of them were used for training and the remaining one for testing. The experiment was repeated 10 times using different 9:1 partitions of the data.

We first ran the SVM classifier on the performance of SVM’s two built-in kernels to make a lower limit and then the \(n\)-gram method of generating sequential features as the baseline. The third run was the case of using the parse-tree kernels using its subset trees. These were all compared to the performance of the case of using the proposed correlation kernel.

\(^3\)Precision measures the number of equally identified sentences as a percentage of the number of sentences identified. Recall measures the number of equally identified sentences as a percentage of the total number of correct sentences.

\(^4\)http://svmlight.joachims.org/
C. Evaluation Result

Our main focus in our experiments was to demonstrate the classification performance, i.e., how well definition sentences are distinguished from elaboration sentences. For such a task in a web-scale setting, however, training time, training data requirements and the amount of data generated for intermediate results are also a concern, especially with machine learning algorithms. Table VI shows the comparison results for the four groups of runs in terms of accuracy (ACC), relative average training time (relAvrTrTime), and relative average disk size (relAvrDskSize) requirement for intermediate results.

For the first group in Table II, we ran two different cases as the lower limit: linear kernel and polynomial kernel based on the bag-of-words approach. The second group refers to the case of using unigrams, bi-grams, and tri-grams as features. Compared to the first group, the n-gram approaches gave a much higher accuracy and reasonable training time although they are relatively simpler than the others in the third and the fourth groups. The only drawback is that they generate significant amount of disk requirements. The third group has two versions of the parse-tree kernel, one using the basic SST-PTK kernel based on lexical items and the other using partial dependency trees that are split into sub-trees as in [22]. SST-PTK-bow has the lowest running time with slightly higher accuracy than the n-gram approaches. Note that these accuracy values are very high already. In fact, SST-PTK-DEP is the second best accuracy in all the experiments although it takes much longer training time than the others in the fourth group, which has three correlation kernel cases corresponding to keyword-collocations pairs, keyword-colligation pairs, and combination of both. They all show very high accuracy, similar to SST-PTK-DEP but faster training time. On the contrary to our earlier observation that syntax alone was not sufficient to distinguish definition from elaboration sentences, the accuracy is extremely high. We attributed its reason to the fact that SST-PTK generate all possible sub-trees and compute similarity scores instead of requiring two syntax (sub)-trees identical. In other words, SST-PTK overcomes the limitations of using syntax patterns alone. Compared to the proposed method, on the other hand, the parse-tree kernel method requires a lot more training data as explained below. Furthermore, the case of using collocations (our proposed method) gave the accuracy comparable with the parse-tree kernel with a much reduced time requirement. This is because of the nature of definition sentences where the existence of content word that co-occur with the keyword is important. By combining both collocations and colligations, we achieved the best accuracy closest to perfection.

### Table III

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>#Docs</th>
<th>#D +#E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>osteoporosis</td>
<td>3608</td>
<td>1000</td>
</tr>
<tr>
<td>2</td>
<td>epilepsy</td>
<td>324</td>
<td>52</td>
</tr>
<tr>
<td>3</td>
<td>thyroid cancer</td>
<td>209</td>
<td>96</td>
</tr>
<tr>
<td>4</td>
<td>hyperlipidemia</td>
<td>390</td>
<td>198</td>
</tr>
<tr>
<td>5</td>
<td>hypertension</td>
<td>238</td>
<td>206</td>
</tr>
<tr>
<td>6</td>
<td>stomatitis</td>
<td>370</td>
<td>108</td>
</tr>
<tr>
<td>7</td>
<td>bronchitis</td>
<td>283</td>
<td>80</td>
</tr>
<tr>
<td>8</td>
<td>cerebral palsy</td>
<td>345</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>diabetes</td>
<td>351</td>
<td>96</td>
</tr>
<tr>
<td>10</td>
<td>arrhythmia</td>
<td>245</td>
<td>106</td>
</tr>
<tr>
<td>11</td>
<td>heart disease</td>
<td>296</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>Alzheimer’s</td>
<td>262</td>
<td>28</td>
</tr>
<tr>
<td>13</td>
<td>depression</td>
<td>339</td>
<td>272</td>
</tr>
<tr>
<td>14</td>
<td>stomach ulcer</td>
<td>299</td>
<td>74</td>
</tr>
<tr>
<td>15</td>
<td>gastritis</td>
<td>158</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>bipolar disorder</td>
<td>334</td>
<td>108</td>
</tr>
<tr>
<td>17</td>
<td>spondylitis</td>
<td>375</td>
<td>220</td>
</tr>
<tr>
<td>18</td>
<td>hemorrhoids</td>
<td>317</td>
<td>80</td>
</tr>
<tr>
<td>19</td>
<td>angina</td>
<td>303</td>
<td>104</td>
</tr>
<tr>
<td>20</td>
<td>conjunctivitis</td>
<td>1159</td>
<td>104</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>9665</strong></td>
<td><strong>3000</strong></td>
<td></td>
</tr>
</tbody>
</table>

### Table IV

<table>
<thead>
<tr>
<th>Experiment</th>
<th>ACC</th>
<th>relAvrTrTime</th>
<th>relAvrDskSize</th>
</tr>
</thead>
<tbody>
<tr>
<td>built-in linear bow</td>
<td>78.01</td>
<td>12</td>
<td>1.013</td>
</tr>
<tr>
<td>built-in poly bow</td>
<td>49.57</td>
<td>1025210</td>
<td>1.013</td>
</tr>
<tr>
<td>1-,2-,3-Gram bow</td>
<td>98.80</td>
<td>218</td>
<td>8.202</td>
</tr>
<tr>
<td>SST-PTK-bow</td>
<td>95.34</td>
<td>1.0</td>
<td>3.419</td>
</tr>
<tr>
<td>SST-PTK-DEP</td>
<td>98.83</td>
<td>184.5</td>
<td>2.097</td>
</tr>
<tr>
<td>CK-COL</td>
<td>98.19</td>
<td>24.7</td>
<td>1.182</td>
</tr>
<tr>
<td>CK-CLG</td>
<td>95.19</td>
<td>20.7</td>
<td>1.0</td>
</tr>
<tr>
<td>CK-COMP</td>
<td>99.32</td>
<td>97.1</td>
<td>2.096</td>
</tr>
</tbody>
</table>

**Table III**

**Data Statistics**

<table>
<thead>
<tr>
<th>Experiment</th>
<th>ACC</th>
<th>relAvrTrTime</th>
<th>relAvrDskSize</th>
</tr>
</thead>
<tbody>
<tr>
<td>built-in linear bow</td>
<td>78.01</td>
<td>12</td>
<td>1.013</td>
</tr>
<tr>
<td>built-in poly bow</td>
<td>49.57</td>
<td>1025210</td>
<td>1.013</td>
</tr>
<tr>
<td>1-,2-,3-Gram bow</td>
<td>98.80</td>
<td>218</td>
<td>8.202</td>
</tr>
<tr>
<td>SST-PTK-bow</td>
<td>95.34</td>
<td>1.0</td>
<td>3.419</td>
</tr>
<tr>
<td>SST-PTK-DEP</td>
<td>98.83</td>
<td>184.5</td>
<td>2.097</td>
</tr>
<tr>
<td>CK-COL</td>
<td>98.19</td>
<td>24.7</td>
<td>1.182</td>
</tr>
<tr>
<td>CK-CLG</td>
<td>95.19</td>
<td>20.7</td>
<td>1.0</td>
</tr>
<tr>
<td>CK-COMP</td>
<td>99.32</td>
<td>97.1</td>
<td>2.096</td>
</tr>
</tbody>
</table>

**Table IV**

**Average Performance of Experiments - 10 cross validation**

Figure 4. Performance comparison between SST-PTK-DEP (dashline) / CK-COMP (straightline) according to different training size

Since acquiring training data is always a concern for supervised machine learning algorithms, researchers have been trying to devise algorithms that minimize the amount of training data. Along the line, we compared the best methods in accuracy, SST-PTK-DEP and CK-COMP (Correlation Kernel - Compositional), for their behavior changes in different training size.
data sizes. Figure 4 shows the accuracy drops measured with reduced amounts of training data from 3,000 sentences to 100 sentences. With only 500 sentences, the accuracy of CK-COMP was similar to that of SST-PTK-DEP using more than 2,000 sentences. That is, although the parse-tree kernel gave a very high performance, it is degraded very rapidly without a sufficient amount of training data. Even with only 200 sentences, which are less than 7% of the total training data, the accuracy of the proposed method is comparable with that of the $n$-gram method.

For further comparison between the proposed correlation kernel and the parse-tree kernel, we measured the total amount of time required from processing the raw text for feature generation to the end of training. Figure 5 shows that SST-PTK-DEP requires about twice as much time as CK-COMP. The X-axis and Y-axis show logarithm scales of the processing times required for different data sizes. The red dotted line is a reference, and the black solid line shows the time requirement ratio between the two methods for different amounts of training data.

![Figure 5. Average time in seconds for the SST-PTK-DEP / CK-COMP](image)

The main effect of the proposed kernel manifests itself when the difference between two feature distributions from definition and exploration sentences, respectively, is not big enough. Our analysis of the feature space created for SVM learning shows that while the features representing the two classes are intermixed for the $n$-gram kernel, the features representing the definition class in the proposed method are distributed quite distinctively from those for exploration class. It clearly indicates that the proposed method plays a useful role as a mapping function.

V. RELATED WORK

There has been a significant amount of research related to recognition/identification of definition sentences. The most popular is a series of work where rules are constructed based on syntactic patterns and templates are predefined to extract definitions. Examples are the effort to find definition sentences from Portuguese text based on morpho-syntactic information as well as words’ inflections [10] and the research on using grammatical rules for identifying definition sentences ([9], [23]).

Still another example is to support a QA system by considering definition sentences as having a template and searching for grammar patterns in Slavic text ([11], [16]). These techniques rely on linguistic knowledge to identify grammatical and word patterns that characterize definitions. Similar approaches but with the motivation of paraphrases also employed a similar technique. In an attempt to find paraphrases, they build templates that represent subjective cases in Japanese and extract definition sentences. ([4], [5]) While the studies of this sort have been trying to improve the rules, they tend to have a limitation of low recall because definition sentences tend to have a variety of syntactic structures.

As an alternative to the rule-based approaches, machine learning has been employed for the same task. There was an attempt to use genetic algorithms [13] where linguistic features were extracted to assign weights to definition sentences. This research also proposed a method of incorporating different characteristics of definition sentences in a compositional way. Instead of using word features, [15] used series of part-of-speech tags to characterize definition sentences. Noting that a sequence of patterns exist in definition sentences, conditional random fields (CRFs) were also used [12]. In order to incorporate domain specificity, [8] attempted to learn the weights of domain-specific terms using bootstrapping.

VI. CONCLUSION AND FUTURE WORK

This research is a first step towards building a knowledge base for healthcare that people can explore and acquire necessary knowledge for various decision makings. Since it is essential to relate symptoms and diseases or health problems for such a knowledge base, we tackled the problem of extracting disease definition sentences that usually contain descriptions of important symptoms. For the purpose of building a knowledge base that can be used for non-experts, we attempted to build a definition sentence classifier that will work for plain-language definitions and tested our method using web forum documents.

The highlight of the proposed method is the use of correlation kernel based on collocation statistics. Unlike the previous approaches where abundant rules and patterns acquired from definition sentence examples are used, we devised a method that rely on collocation statistics that are language independent. In addition, this approach addresses a notable problem in using machine learning techniques for big data processing. It achieves a very high accuracy level by using a very small amount of training data, which is an order of magnitude difference compared to the data required by the parse-tree kernel method for a similar performance.

Given the accuracy approaches to 100%, we are ready to tackle the problem of extracting symptoms and related entities from definition sentences so that they become part of the skeletons of the knowledge base we are trying to construct. At the same time, we are going to explore a different way of constructing gold standard for training and testing, to ensure that there are other forms of definition and elaboration...
sentence candidates that are not extractable using the heuristics we used for the experiments.

ACKNOWLEDGMENT

This work was supported by the IT R&D program of MOTIE/KEIT. [10044970, Development of Core Technology for Human-like Self-taught Learning based on Symbolic Approach]

REFERENCES

[22] L. Qian, G. Zhou, Dependency-directed Tree Kernel-based Protein-Protein Interaction Extraction from Biomedical Literature, IJCNLP 2011: 10-19