Abbreviation Disambiguation Using Semantic Abstraction of Symbols and Numeric Terms

Sa Kwang Song
Information and Communications University
Daejeon
smallj@icu.ac.kr

Yun JIN
Chungnam National University
Daejeon
wkim@cnu.ac.kr

Sung Hyon Myaeng
Information and Communications University
Daejeon
myaeng@icu.ac.kr

***Abstract*-We propose an abbreviation disambiguation approach that utilizes semantic representation of symbols and numeric terms as well as the words in clinical documents. While majority of related works treats symbols and numeric words as stopword, we show that they play an important role especially in coarse-grained documents such as CDA (Clinical Document Architecture) documents, which contain a lot of jargons, symbols, and abbreviations written by doctors. For abbreviation disambiguation task using a classifier, we compared several variations of our approach with a traditional bag-of-words method. The results show that the system using semantic abstraction of symbols and numeric terms can improve the accuracy from 87.9% to 92.6% when a SVM classifier is used.**

I. INTRODUCTION

Various text-based applications such as information retrieval, text classification, information extraction, question answering focus on document processing at a lexical, syntactic, or even semantic level. Documents generally contain unclear words including multivocal words and abbreviations that make it difficult to represent the document accurately.

Especially in the medical domain, clinical documents written by doctors tend to have very different characteristics compared to scholarly articles or papers. A general phenomenon across any kind of medical documents is that they contain many specialized medical words which are hardly found in those in other areas. In particular, clinical documents are not well structured linguistically because doctors write their medical opinions or diagnoses for themselves or for their associates who learn their characteristics over time; they are not for general public's consumption. For example, it is hard to find sentence boundaries since doctors usually ignore punctuation symbols such as periods and commas and do not always follow the basic sentence structure such as *Subject+Verb*. One of the unique problems is to correctly interpret abbreviations that are convenient for doctors to express ideas and opinions quickly and briefly but often carry several meanings.

Abbreviation is a shortened form of a word or a sequence of words. The original word or sequence of words is called a long form of abbreviation. Abbreviation disambiguation means to choose the correct long form, often based on its context. Medical documents in the form of Clinical Document Architecture (CDA) especially contain many abbreviations that are frequently used by doctors.

Abbreviations usually have following three forms:

- Most of abbreviations consist of capital words, which are extracted from the first letter of each words, such as "US" is an abbreviation of "United State".
- Some abbreviation use phonetic features, likely, "A4U"is an abbreviation of "all for you".
- Some other abbreviation also used other symbol, and image, such as "I love you" is abbreviation to "I love you" or "I♥U", "less than" is abbreviate to "<=".

In this paper, we focus on the abbreviation disambiguation where the task is to assign an appropriate sense out of several candidates to an occurrence of an abbreviation in a given context. There has been research on this problem in the biomedical as well as computer science. The most popular approach is to solve it using contextual information such as neighboring words or phrases in well-formed documents such as Medline abstracts or thesis papers [7]. Document analysis with those containing grammatically well-formed sentences is relatively easier than with CDA documents because most deep natural language processing techniques have been developed for grammatically correct sentences

In order to deal with the word sense disambiguation problem, many algorithms have been proposed including knowledge or dictionary based, corpus based, and machine learning (supervised or unsupervised) based algorithms. Even though supervised learning algorithms are the most prominent ones, they need a training set for which laborious human efforts are required. Recently, the Support Vector Machine (SVM) algorithm is frequently used for classification tasks.

To decrease the burden of developing training data, we employ an automatic full-name extraction method that finds the long form of an abbreviation in the entire document collection and makes use of them as training data.

This paper continues with a description of related work, a brief explanation about the corpus and the abbreviation set, our methodology including a machine learning method, the training and test set, and the feature extraction methods.

II. RELATED WORK

Word sense ambiguities make an application less effective since they are propagated to the performance of the upper level application such as information extraction,

0-7803-9361-9/05/$20.00 ©2005 IEEE

summarization, question answering, etc. Although much research has been done, little has been done in the medical domain.

YU et al. [7] developed an automatic abbreviation disambiguation system in Medline abstracts using SVM machine-learning technique and One Sense per Discourse Hypothesis method. Their experiments show that their method achieves an accuracy of about 84%.

Lee et al. [15] also tried to utilize SVM algorithm both for disambiguating general English words and for translating an ambiguous English word into its Hindi equivalent. For SVM features, they have made use of unigrams, collocations, parts of speech and semantic relations. The results that they obtained for general English corpus were better than those for the translation task.

In addition, Ngai et al. [16] proposes a supervised approach to semantic role labeling by posing it as a classification task of assigning the words to categories from the FrameNet ontology. They make use of various lexical and syntactic features from FrameNet as well as some extracted features, with machine learning methods like Boosting, SVMs, Decision Trees etc. and their combined models.

Liu et al. [4] compared the performance of various classifiers on two medical data sets and one general English data set. Their classifiers included the traditional decision lists, their adaptation of the decision lists, the naïve Bayes classifier and a mixed learning approach they developed. Their features included local co-occurring words, collocations and some derived features like parts of speech and semantic categories in various window sizes around the word of interest.

Even though their works performed quite well and one of them reaches up to 95% accuracy, it is not clear whether their methodologies can be applied to unconventional documents, such as CDA documents that contain patient's symptoms, prescriptions, treatments and so on. One of the important characteristics of CDA documents is that they have a large amount of symbols and numeric terms.

R/O COPD, R/O pneumothorax로 check한 chest PA상 hilar enlargement 있고 mass shadow 있어 R/O lung cancer로 w/u 위해 내원함 PMHx> DM/HT/Tb/Hepatic ds(-/-/-/-) Social Hx> smoking(+): 1-2pack/day * 50yr alcohol(+): heavy S/R> G/W(+), E/F(+), f/c(-/-), c/s/r(+/+/-), HA/Dz(-/-) wt. change(+): 10kg loss, dyspnea(-), chest discomfort(-) indigestion(-), epi. soreness(-) A/N/V/D/C(+/-/-/-/-), H/M/H(-/-/-) abdominal pain(-), dysuria(-), foamy urine(-) 123-4.7-90-22 BUN/Cr 53/11.7 CBC 6160-6.9-246k peritosol cell (-) prot 58 LD 3 Iron/TIBC 121/207 ferritin 372 CRP 0.1

Fig.1. Snippet of a CDA document

III. DATA

A. CDA Document

The CDA is an HL7 (Healthcare Level 7) standard for the representation and machine processing of clinical documents in a way which makes the documents both human readable and machine processable, and guarantees preservation of the content by using the eXtensible Markup Language (XML) standard. It is a useful and intuitive approach to management of documents which make up a large part of the clinical information processing area [11].

The CDA documents include a large amount of symbols and numeric words compared to general documents in order to represent the performance of treatments or the states of patients. The percentage of symbols and numeric terms in CDA documents appears to be up to 10% based on the term statistics we calculated using the CDA document set, which is much higher than that those in other domains. Fig. 1 is a snippet from a CDA document.

As can be seen in the snippet, abbreviations occurs very frequently in CDA documents compared to other documents such as Medline abstracts and thesis papers which have been a popular corpus for WSD domain. It also contains many symbols and numeric terms that are usually treated as stop words and removed in traditional text processing. Since they play an important role in the CDA documents, however, we have to deal with them in appropriate ways so as to find the exact sense of an abbreviation as well as to understand documents more clearly. Our corpus consists of 15,618 documents that were provided by Seoul National University Hospital (SNUH) for research purposes.

B. Abbreviation Set

The abbreviation set used in our work is constructed with help of Department of Biomedical Engineering in SNUH. It contains around 310 frequently cited abbreviations and their average senses per abbreviation are 3.267. Fig. 2 shows the number of senses per abbreviation. The number of senses per abbreviation varies from 2 to 8. However, most of them are ranged from 2 to 4. In table I, there are examples of abbreviations with their full names. The rightmost column shows the number of instances that we have gathered both automatically and manually at SNUH.

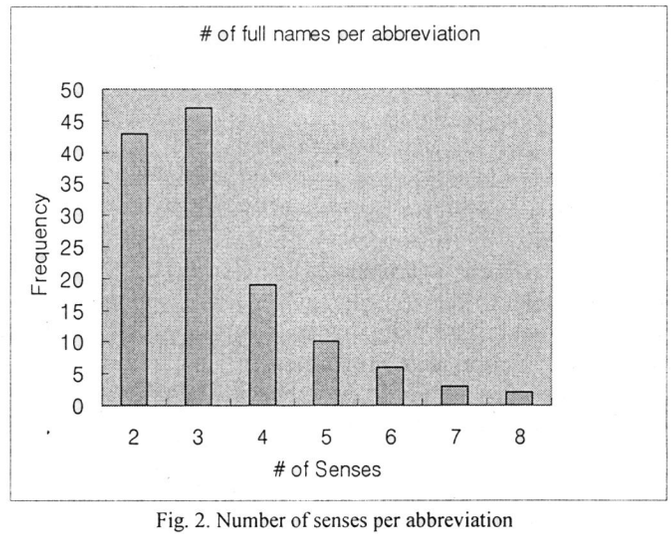

Fig. 2. Number of senses per abbreviation

TABLE I
EXAMPLE OF ABBREVIATIONS AND THEIR CORRESPONDING FULL NAMES; COUNT MEANS THE NUMBER OF FULL NAMES, AND N MEANS THE NUMBER OF OCCURRENCES IN THE CDA CORPUS.

Abbreviation	Count	Full Name	N
ACA	3	Adenocarcinoma anterior cerebral artery anterior communicating artery	1364
AF	4	Atrial Fibrillation Atrial flutter abnormal frequency acid-fast	418
AS	5	apgar score activated sleep anal sphincter ankylosing spondylitis aortic stenosis	109
DS	5	dental surgery dead air space dead space deep sleep down's syndrome	220
FC	7	Fronto temporal Free thyroxine Fallot tetralogy function test full term foot flexor tendon	1038
IMC	6	Internal medicine pulmonary Inosine 5 monophosphate idiopathic myeloid proliferation impression improved important	2950

IV. METHODOLOGY

A. SVM Classifier

We apply SVM as a classification method to resolve the abbreviation ambiguity which is a state of the art supervised machine-learning technique proposed by Vapnik and is based on Structured Risk Minimum Principle. By the principle, when training a classification model, the aim of the learner is to optimize not just the error rate on the training data set, but also the ability of the model for prediction, and the ability depends on concept VC-dimension. Following the Structured Risk Minimum Principle, training a SVM is summed up as finding optimal classifying hyperplane that has the largest margin. The margin is defined as the distance from the hyper plane to the closest training examples. The SVM is being applied in many areas such as text classification, word sense disambiguation, and has showed many advantages over the other supervised machine-learning methods. [7] We use SVM^{light} as an implementation of SVM algorithm [12].

B. Training and Test Data

Because SVM is a supervised learning method, we need to build up a tagged training corpus which requires laborious manual tagging. Therefore, we attempted to automatically found all occurrences of the full name that correspond to each abbreviation used in our corpus so as to reduce the effort of building training set. Since there is no guarantee that the training set can cover all abbreviations. we had four graduate students from the SNUH find appropriate abbreviations and select an appropriate sense for each abbreviation in a context. At last, we gathered averagely 46.5 and 209.3 tagged contexts by hand and by program respectively. Table II shows the number of tagged instances for training and test set. Two third of them are used for SVM training purpose and one third for testing purpose.

TABLE II
THE NUMBER OF OCCURRENCES IN THE CDA CORPUS GATHERED IN AUTOMATIC AND MANUAL WAY.S

	Manual Tagging	Automatic Tagging	Sum
Training Set	31	141.5	172.5
Test Set	15.5	67.8	83.3
Total	46.5	209.3	255.8

C. Feature Extraction

Resolving sense ambiguity usually requires the context information which is described by a set of neighboring words. We therefore utilize a set of adjacent terms as the context information. The neighboring terms are basically represented as a vector of frequency. The vector forms like below.

$$(L, F_1{:}V_1, F_2{:}V_2, \ldots, F_N{:}V_N),$$

Where L represents which long form is being used in this occurrence, the $F_1, \ldots, F_N$ means terms that describe the abbreviation context, and $V_1, \ldots, V_N$ stands for the value of each feature. The results depend on how and what features are taken. In the following sub-sections, we introduce four methods for constructing feature vector.

(1). Bag-of-words

A context containing an abbreviation consists of a set of words which give clues for disambiguation. Each word is very basic but important clue for guessing the meaning of an abbreviation. Therefore, this method treats the neighboring words as a bag-of-words and represents them as a vector with term and its frequency.

Suppose we have a sentence as follows.

$$\text{“}W_1\ W_2\ W_3\ W_2\ W_1\ \textbf{\textit{ABBR}}\ W_3\ W_5\ W_6\ W_1\ W_7\text{”}$$

Where the W_i stands for a word and ABBR a full name of an abbreviation. So, the feature vector (FV) is represented below with the number of occurrences, and then normalized.

$FV=(\boldsymbol{ABBR}, W_1{:}3, W_2{:}2, W_3{:}2, \ldots W_7{:}1)$

In order to investigate how the number of neighboring words affects the performance, we observe the accuracy changes while modifying the window size from 10 to 30.

(2). Weighting Scheme

The performance of the SVM classifiers much depends on how word weight is extracted in a context. Some research used just the existence of a word within a predefined window, but others applied the *tf*idf* method. We basically utilize the *tf-idf* idea but modify it as follows.

$$WGT(t_i) = \frac{TF(t_i)}{\arg max(TF(t_1)...TF(t_N))} * \log(\frac{N_s}{n_s})$$

where $TF(t_i)$ be the raw frequency of the term t_i, N_s be the total number of senses, and n_s be the number of senses in which the index term t_i appears.

(3). Semantic Tags

Since the CDA documents are written by doctors who are not necessarily good at typing and were trained to use many jargons, they contain many symbols, numerical terms, and abbreviations. In our CDA document corpus, around 10% of words are symbols, numeric expressions, or both. If we ignore them as in the general text processing, we may loose important clues since many neighboring terms are symbols or numbers. Fig. 3 shows some of the examples of symbols and numeric expressions that are frequently found in our CDA documents.

T/RT(- /-)
Dorsalis pedis a. (++/++)
Carotid bruit(+ suspicious/-)
F/U/D(++/+/-)
Hx(+ : lt. MRM)
intact to pain DTR : ++,++/++,++ A/C(-/-), Babinski(-/-)
155-5.5/3.2-0.5-65-16/18
15,480-9.0-342K
4700-16.5-52.8-144K
U/A : alb 3+, glu 1+, Bld 2+
11.5%
FEV1 2.02(67%) 2.11 1.92 1.74 1.77 1.80 1.50
0.5 tab tid ++ 1 일 3회 * 14일
95/70 - 72 - 18 - 37.3 ℃
LNE(-/-) V/E(-/-) T/E(-)

Fig. 3. The example of symbols and numeric terms

We assume that symbols and numeric expressions play an important role in resolving sense ambiguity in our text. Therefore, we introduce a semantic pattern based term abstraction method for symbols and numeric expressions, which categorizes similar numeric patterns into a semantic concept. That is, semantically identical set of terms is tagged with a same conceptual annotation shown in table III.

TABLE III
EXAMPLES OF SEMANTIC PATTERN BASED ON REGULAR EXPRESSION

Pattern Rule	Tag	Meaning
\d+(\.\d+)?	{num}	Number
\d+	{int}	Integer
\s{0,2}	{s}	Space
([\x80-\xff][\x80-\xff])+	{kor}	Korean
[ ]*[0-2]?[0-9]:[0-6]?[0-9]	TIME	Time
{s}{num}({s}-{s}{num})*{s}((\(.+?\))|(.*%))	ABGA	ABGA related pattern
{s}{num}{s}({s}-{s}{num}{s})*(/{s}{num}{s}({s}-{s}{num}{s})*){2,5}	PERF	Performance
[a-z]+{s}[a-z0-9]+{s}{int}{s}일{s}{int}회{s}*{s}{int}일	DRUG	Dosage
\([-+]+(/[-+]+)*\)	EFCT	Effect

In table III, the pattern rules are described by regular expression, and the notations within braces such as '{s}', '{num}', '{int}', and '{kor}' are replacements of a set of regular expressions for simplicity of later use. For example of semantic pattern abstraction, "(++/--/++)" will be tagged with "EFCT" by the rule "\([-+]+(/[-+]+)*\)", which usually represents a treatment or procedure effect.

V. EXPERIMENTS AND RESULTS

We designed a baseline system and two comparison systems to show the effectiveness of using semantic abstraction of symbols and numeric words. An implementation of SVM, SVM^{light}, is applied as a classifier. The CDA documents have been processed by Korean and English POS taggers and stopword elimination process which does not remove symbols and numerical terms except when we use keywords only as features.

A. Experiment using Keyword Only

The baseline system uses the term itself as a feature and its frequency as a feature value in a predefined window size. The stopwords including symbols and numeric terms are eliminated in general text processing. The SVM^{light} provides four kernel methods such as *linear*, *polynomial*, *RBF* (Radial Basis Function), and *tanh* kernels. Among them, the linear kernel based method performs best when we compared the four with window size 30 although the polynomial kernel gave a similar performance.

TABLE IV
KERNEL COMPARISON WITH WINDOW SIZE 30

Kernel Method	Precision
Linear	0.921947
Polynomial	0.912279
RBF	0.878172
Tanh	0.835285

Table IV shows the result precision values of baseline systems using the linear kernel method with a fixed windows size. When the linear kernel with windows size 30 is used, it performs better than any other kernel methods. Additionally, we compared the precision values of abbreviation disambiguation methods corresponding to the varying window size ranging from 5 to 30. Fig. 4 shows the results. In general, as the size of window is enlarged, the precision is increased.

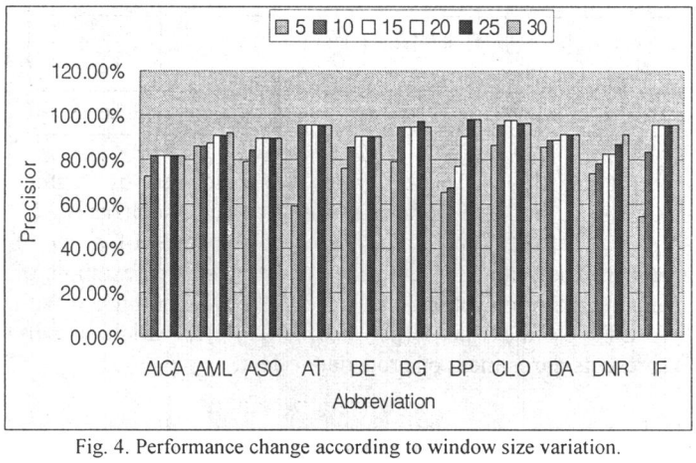

Fig. 4. Performance change according to window size variation.

B. Experiment using Keyword + Semantic Abstraction

This experiment shows how the precision increases when we add the semantic abstraction information about symbols and numeric terms to the keyword features. The third column in Table V notes the performance increase compared to the baseline system. Although the increase ratio of performance decreases as window size enlarges, this experiment shows it's effectiveness in case there is a little clue for sense disambiguation. Considering that most WSD works uses less number of context words because of the efficiency, this result is of enough worth.

C. Experiment using Keyword + Semantic Abstraction + Term Weighting

This experiment proves the importance of weighting scheme, which is added to the previous experiment. In table V, we could get improvement of about %1 compared to the case with semantic abstraction, and a maximum of 5% compared to the baseline case. We feel that the amount of improvement is significant enough because in general it is difficult to increase precision in the 90% range while easier in the 70-80% range.

Fig. 5 explicitly shows the performance comparison among three methods. It appears that our semantic abstraction based method helps to resolve abbreviation ambiguity problem especially when the amount of context information is not much. The weighting scheme also improves the precision in a small window size setting.

TABLE V
PRECISION COMPARISONS OF THE THREE APPROACHES: K+S MEANS KEYWORD+SEMANTIC ABSTRACTION METHOD, AND K+S+W APPENDS THE WEIGHING SCHEME TO K+S METHOD. THE PERCENTAGE IN PARENTHESES STANDS FOR IMPROVEMENTS COMPARED TO THE BASELINE METHOD.

Window Size	Keyword (Baseline)	K+S	K+S+W
10	0.879031	0.915624 (4.16%)	0.923505 (5.06%)
20	0.904525	0.927818 (2.58%)	0.931622 (3.00%)
30	0.921947	0.927309 (0.58%)	0.934644 (1.01%)

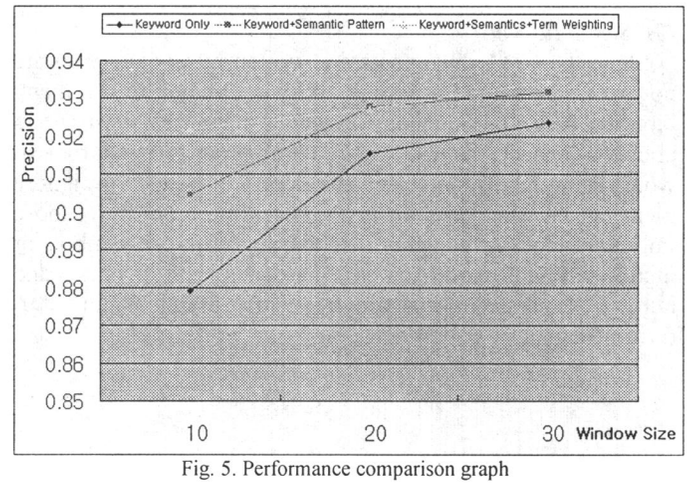

Fig. 5. Performance comparison graph

VI. CONCLUSION

While most of text processing techniques treat symbols and numeric expressions as stop words, we argued that they play an important role in sense disambiguation of abbreviations and proved it by a set of experiments. Especially, we showed that the semantic abstraction of frequently used patterns helps to improve the disambiguation performance. For future work, we are going to utilize the UMLS (Unified Medical Language System) meta-thesaurus which helps the matching a word or phrase with high level biomedical concept.

ACKNOWLEDGMENT

This study was supported by a grant of the Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea (02-PJ1-PG6-HI03-0004).

REFERENCES

[1] Joshi et al. "Supervised Word Sense Disambiguation in the Medical Domain using Support Vector Machines." JAMIA, 2004.

[2] Hongfang Liu, et al. "A Study of Abbreviations in MEDLINE Abstracts," AMIA 2002.

[3] Hongfang Liu, et al. "Automatic Resolution of Ambiguous Terms Based on Machine Learning and Conceptual Relations in the UMLS," JAMIA 2002.

[4] Hongfang Liu et al., "Multi-aspect comparison study of supervised word sense disambiguation", JAMIA 2004

[5] Yaakov HaCohen-Kerner et al. "Baseline Methods for Automatic Disambiguation of Abbreviations in Jewish Law Documents," EsTAL 2004.

[6] Antonio Molina et al. "A Hidden Markov Model Approach to Word Sense Disambiguation," LNCS 2002

[7] Zhonghua YU et al. "Automatic Resolution of Ambiguous Abbreviations in Biomedical Texts using Support Vector Machines and One Sense Per Discourse Hypothesis," SIGIR 2003.

[8] Regular Expression HOWTO Homepage by A.M. Kuchling: http://www.amk.ca/python/howto/regex/

[9] Christopher J. C. Burges et al. "A Tutorial on Support Vector Machines for Pattern Recognition," Kluwer Academic Publishers, Data Mining and Knowledge Discovery, 2, 121-167, 1998

[10] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. "Choosing multiple parameters for support vector machines." Machine Learning, 46(1-3):131-159, 2002.

[11] What is CDA?: http://www.hl7.org.au/CDA.htm#CDA

[12] SVMlight: http://svmlight.joachims.org/

[13] Unified Medical Language System (UMLS) : http://www.nlm.nih.gov/research/umls/

[14] Vladimir N. Vapnik, "The Nature of Statistical Learning Theory." Springer, 1995.

[15] Yoong Keok Lee et al., "Supervised Word Sense Disambiguation with Support Vector Machines and Multiple Knowledge Sources," SENSEVAL-3, ACL, 2004

[16] Grace Ngai et al., "Semantic Role Labeling with Boosting, SVMs, Maximum Entropy, SNOW, and Decision Lists," SENSEVAL-3, ACL, 2004

[17] Seoul National University Hospital : http://www.snuh.org/