A Learning Approach for Word Sense Disambiguation in the Biomedical Domain

Hisham Al-Mubaid*
University of Houston-Clear Lake
Houston, TX, 77058, USA
hisham@uhcl.edu

Sandeep Gungu
University of Houston-Clear Lake
Houston, TX, 77058, USA
gungus@uhcl.edu

Abstract
Word sense disambiguation, WSD, task has been investigated extensively within the natural language processing domain. In the biomedical domain, word sense ambiguity is more widely spread with bioinformatics research effort devoted to it is not commensurate and is allowing for more development. In this paper, we present and evaluate a machine learning based approach for WSD. The main limitation with supervised methods is the requirement for manually disambiguated instances of the ambiguous word to be used for training. However, the advances in automatic text annotation and tagging techniques with the help of the plethora of knowledge sources like ontologies and text literature in the biomedical domain will help lessen this limitation. Our approach has been evaluated with the benchmark dataset NLM-WSD with three settings. The accuracy results showed that our method performs better than recently reported results of other published techniques.

* *corresponding author: Hisham@uhcl.edu*

1. Introduction

If a word has more than one sense then the task of determining the sense of that word in a given text is defined as word sense disambiguation. Within the natural language processing (NLP) domain, the word sense disambiguation (WSD) problem has been studied and investigated extensively over the past few decades [1, 2]. In the biomedical domain, WSD is more widely spread in the biological and medical texts and sometimes with more severe consequences. However the research efforts in the biomedical domain to solve WSD are not proportional to the extent of the problem. As an example, in the biomedical texts, the term "*blood pressure*" has three possible senses according to the Unified Medical Language System (UMLS) [7] as follows: *organism function*, *diagnostic procedure*, and *laboratory or test result.* Word sense disambiguation has important applications and uses in the text mining, information extraction and information retrieval systems [1-3]. It also is considered a key component in most intelligent knowledge discovery and text mining applications.
The main classes of approaches of word sense disambiguation includes supervised methods and unsupervised methods. The supervised methods rely on training and learning phases that require a dataset or corpus that includes manually disambiguated instances to be used to train the system [17, 18]. The unsupervised methods, on the other hand, are based on knowledge sources like ontology (UMLS) or text corpora [2, 3, 5, 8]. The approach presented in this paper is a supervised approach. In this paper, we present and evaluate a supervised method for biomedical word sense disambiguation. The method is based on machine learning and uses feature extraction techniques in constructing feature vectors for the words to be disambiguated. We evaluated the method with the NLM-WSD benchmark corpus from the biomedical domain. The evaluation results proved the competitiveness of our method as it outperforms some recently published techniques including supervised techniques.
Related Work:- A number of methods have been presented in the bioinformatics literature for biomedical word sense disambiguation [1-3, 6, 8, 9]. In [9], Humphrey et al. (2006) uses UMLS as knowledge source for assigning the correct sense for a given word. They used journal descriptor indexing of the abstract containing the term to assign a semantic type from UMLS metathesaurus [7, 9]. Agirre et al. (2010) present a graph-based method which is considered unsupervised but relies on UMLS [2]. The concepts of UMLS are

represented as a graph and WSD is done using personalized page rank algorithm [2]. The work in [1] uses supervised learners with linguistic features extracted from the context of the word in combination with MeSH terms for disambiguation. In [3], Jimeno-Yepes and Aronson (2010) presented a review and evaluation of four approaches that rely on UMLS as the source for knowledge for disambiguation.

2. A Method for WSD

A word sense disambiguation method is an algorithm that assigns the most accurate sense to a given word in a given context. Our method is a supervised method that requires a training text that contains manually disambiguated instances of the word. The method is based on a word classification and disambiguation technique that we have proposed in a previous work [4]. It relies on representing the instances of the word to be disambiguated, *w*, as a feature vector and the component of this vector are neighborhood context words in the training instances. In the context of the target word, *w*, we select the word with the high *discriminating* capabilities as the components of the vectors. We use the manually disambiguated instances in the training corpus as labeled training examples. The classifiers will then be used to disambiguate unseen and unlabeled examples in the testing phase. That is, during the training phase, the constructed feature vectors of the training instances will be used as labeled exampled to train classifiers. The classifier (*model*) will be then used to disambiguate new, unseen, and unlabeled examples in the testing phase. One of the main strength of this method is the features are selected for learning and classification.

Feature Selection:- The features selected from the training examples have great impact on the effectiveness of the machine learning technique. Extensive research efforts have been devoted to feature selection in machine learning research [10-13]. The labeled training instances will be used to extract the word features for the feature vectors.

Suppose the word w_x has two senses s_1, s_2, and let the set C_1 be the set of w_x instances labeled with s_1 and C_2 contains instances of w_x labeled with sense s_2. So each instance of w_x labeled with sense s_1 or s_2 (*i.e.*, in the set C_1 or in the set C_2) can be viewed as:

$$p_n \ldots p_3\, p_2\, p_1 < w_x; s_i > f_1\, f_2\, f_3 \ldots f_n$$

where the words p_1 , p_2 ,…. , p_n and f_1 , f_2 , ,….., f_n are the context words surrounding this instance, and *n* is the *window size*. Next, we collect all the context words p_i and f_i of all instances in C_1 and C_2 in one set *W* (*s.t.* $W = \{w_1 ,w_2 , \ldots ,w_m\}$). Each context word $w_i \in W$ may occur in the contexts of instances labeled with s_1 or with s_2 or combination and in any distribution. We want to determine that, if we see a context word w_i in an ambiguous instance, to what extent this occurrence of w_i suggests that this example belongs to C_1 or to C_2. Thus, we use as features those context words w_i that can highly discriminate between C_1 and C_2. For that, we use feature selection techniques such as *mutual information* (MI) [11, 12] as follows. For each context word $w_i \in W$ in the labeled training examples, we compute four values *a*, *b*, *c*, and *d* as follows:

a = number of occurrences of w_i in C_1
b = number of occurrences of w_i in C_2
c = number of examples of C_1 that do not contain w_i
d = number of examples of C_2 that do not contain w_i

Therefore, the *mutual information* (MI) can be defined as:

$$\mathrm{MI} = \frac{\mathrm{N} * \mathrm{a}}{(\mathrm{a}+\mathrm{b}) * (\mathrm{a}+\mathrm{c})} \qquad \ldots..(1)$$

and *N* is the total number of training examples. Moreover, we define another method, *M2*, for selecting the words as features to be included in the feature vectors as follows:

$$\mathrm{M2} = \frac{\mathrm{a}+\mathrm{d}}{\mathrm{b}+\mathrm{c}} \qquad \ldots..(2)$$

Then, *MI* (or *M2*) value is computed for all context words $w_i \in W$. Then the context words w_i are ordered based on their MI values and the top *k* words w_i with highest MI values are selected as features. In this research, we experimented with *k* values of 200 and 300. With *k*=100, for example, each training example will be represented by a vector of 100 entries such as the first entry represent the context word w_i with the highest MI value, the second entry represents the context word with the second highest MI value and so on. Then for a given training example, the feature vector entry is set to 1 if the corresponding feature (*context*) word occurs in that training example and set to 0 otherwise. Table 1 shows the top 10 context words with the ten highest MI values for the ambiguous word '*cold*' in the NLM-WSD benchmark corpus explained in Section 3. These 10 words will be used to compose the feature vectors for training or testing examples of the terms to be disambiguated. For example, this feature vector

[0 0 1 0 1 0 0 1 0 0]

represents an example containing the 3rd, 5th and 8th feature words in the context of the word within certain window size.

The Training Step:- From the labeled training examples of the word we build the feature vectors using the top context words selected by MI or M2 as features. After that, we use the support vector machine (SVM) [14] as the learner to train the classifier using the training vectors. SVM is one of the most successful and efficient machine learning algorithms, and is well founded theoretically and experimentally [4, 5, 10, 14]. The applications of SVM are abound; in particular, in NLP domain like text categorization where SVM proved to be the best performer. We use *SVM-light* (svmlight.joachims.org) implementation with the default parameters and with the *RBF* kernel.

The Disambiguation Step:- in the testing step we want to disambiguate an instance w_q of the word w. We construct a feature vector V_q for the instance w_q the same way as in the training step. Then we apply the classifier on V_q to classify it (assign w_q) to one of the two senses.

3. Evaluation and Experiments

Dataset:- We used the benchmark dataset NLM-WSD for biomedical word sense disambiguation [15]. This dataset was created as a unified and benchmark set of ambiguous medical terms that have been reviewed and disambiguated by reviewers from the field. Most of the previous work on biomedical WSD uses this dataset [1-3]. The NLM-WSD corpus contains 50 ambiguous terms with 100 instances for each term for a total of 5000 examples. Each example is basically a *Medline* abstract containing one or more occurrences of the ambiguous word. The instances of these ambiguous terms were disambiguated by 11 annotators who assigned a sense for each instance [15]. The assigned senses are semantic types from UMLS. When the annotators did not assign any sense for an instance then that instance is tagged with '*none*'. Only one term '*association*' with all of its 100 instances were annotated *none* and so dropped from the testing.

Context words *wi*
import
understand
ischemia
reperfus
respons
stor
arteri
attempt
repres
quantit

Table1: Context words with the top MI values for the ambiguous word '*cold*'

Text Preprocessing:- On this benchmark corpus, we have carried out some text preprocessing steps:

– Converting all words to *lowercase*.

– Removing *stopwords*: removing all common function words like '*is*' '*the*' '*in*', ..etc.

– Performing word *stemming* using *Porter* stemming algorithm [16].

Moreover, unlike other previous work, words with less than 3 or more than 50 characters are not ignored currently (unless dropped by the stopword removal step). Also words with parentheses or square brackets are not ignored and Part-of-speech is not used.

After the text preprocessing is completed, for each word we convert the instances into numeric feature vectors. Then we use SVM for training and testing with 5-fold cross validation such that 80% of the instances are used for training and the remaining 20% are used for testing and this is repeated five times by changing the training-testing portions of the data. The accuracy is taken as the mean accuracy of the five folds and the accuracy is computed as

$$Accuracy = \frac{no.of\ instances\ with\ correct\ assigned\ senses}{total\ no.of\ tested\ instances}$$

We also use the *baseline* method which is the most frequent sense (*mfs*) for each word.

Experiments:- initially we evaluated our WSD method with all the 49 words (excluded *association* as mentioned previously) such that, a word is included in the evaluation only if it has at least two or more senses with each sense has at least two instances annotated with it. This lead to a total of 31 words tested in this evaluation and 18 words were dropped because they do not have at least two instances annotated for each one of two senses. For example, the word '*depression*' has two senses: *Mental or Behavioral Dysfunction* and *functional concept*. Out of the 100 instances of *depression*, 85 instances are tagged with the first sense and remaining 15 instances are tagged with '*None*' (*i.e.*, no instances tagged with a second sense) and so it was excluded in this evaluation. Likewise the word '*discharge*' was not tested as has only one instance tagged with the first sense, 74 instances tagged with the second sense, and 25 instances tagged with *None*. We used *k=200* and the *window* size is 5. The accuracy results of this first evaluation (*EV1*) are shown in Table 2. The detailed results of this evaluation are included in Table 3.

In the second evaluation (*EV2*) and third evaluation (*EV3*) we changed the parameter and the word/features selection formula. In EV2 we set *k=300* and window size is still 5. In EV3, we kept *k=300*, window=5, and changed the word/feature selection formula to *M2* defined in equation (2). Table 3 contains the results of EV2 and EV3. To judge on performance of our method and compare our results with similar techniques, we included several reported results from three recent publications from 2008 to 2010 [1, 2, 3] with our results in Table 4.

4. Discussion and Conclusion

The main weakness of the supervised and machine learning based methods for WSD is their dependency on the annotated training text which includes manually disambiguated instances of the ambiguous word [2, 4]. However, over the time, the increasing volumes of text and literature in very high rates and the new algorithms and techniques for text annotation and concept mapping will alleviate this problem. Moreover, the advances in ontology development and integration in the biomedical domain will facilitate even more the process of automatic text annotation.

In this paper we reported a machine learning approach for biomedical WSD. The approach was evaluated with a benchmark dataset, NLM-WSD, to facilitate the comparison with the results of previous work. The average accuracy results of our method, compared to some recent reported results (Table 4), are promising and proving that our method outperforms those recently reported methods. Table 4 contains the results for 11 methods: baseline method (mfs), our method (last column), and 9 other methods from recent work published in 2008 to 2010 (from Refs [1][2][3]). The average accuracy of our method is the highest (90.3%) and the closest one is NB (86.0%). Our method also outperforms all 10 other methods in 12 out of 31 words followed by NB which outperforms the rest in 7 words.

	Accuracy
Fold 1	0.912
Fold 2	0.931
Fold 3	0.917
Fold 4	0.897
Fold 5	0.862
Average	**0.903**

Table 2: Accuracy results of the first evaluation, EV1, where each sense has to have at least two instances tagged with it.

Stevenson et al. (2008) in their paper [1] report extensive accuracy results of their method (we call it *Stevenson-2008*) along with four other methods including Joshi-2005 and McInnes-2007, with various combinations of words from *NLM-WSD* corpus used for testing. For example, Joshi-2005 tested their system on 28 words (out of the whole set 50 words) and other techniques used 22 words, 15 words, or the whole set [1]. In Table 4, the results of the three methods (Joshi-2005, McInnes-2007, and Stevenson-2008) are taken from Stevenson et al. (2008) [1]. These three methods are supervised methods and used various machine learning algorithm and wide sets of features. For example, Stevenson-2008 used linguistic features, CUI's, MeSH terms, and combination of these features. They employed three learners VSM (vector space model), Naïve Bayes (NB) and SVM. The results included in Table 4 are their best results with VSM and (linguistic + MeSH) features [1]. The method of Joshi-2005 uses five supervised learning methods and collocation features while McInnes-2007 uses NB [1].

Our evaluation is done on 31 words (*as explained in Sec 3*). We obtained the results of the other methods on these 31 words from the references shown in Table 4 to allow for direct comparison. The best result reported in their paper is 87.8% using all words with VSM model and for McInnes 85.3% also with the whole set [1]. The best result of Stevensons-2008 for subsets was 85.1% using a subset of 22 words defined by Liu et al. (2004) [1]. The results of the three methods (Single, Subset, Full) in Table 4 are taken directly from Agirre et al. (2010) [2]. As shown in Table 4, the average accuracy of these three methods (68.8%, 59.7%, 63.5%) on the 31 words are significantly lower than our method (90.3%). Also the average accuracy of their method on the whole set (65.9%, 63.0%, and 65.9%); we note that their method is unsupervised and does not require tagged instances [2]. In another work, Jimeno-Yepes and Aronson (2010) evaluate four unsupervised methods on the whole NLM-WSD set [3] as well as NB and combination of the four methods. The accuracy of the four methods ranges from 58.3% to 88.3% (NB) on the whole set, and *NB* found to be the best performer followed by *CombSW* (76.3%) [3]. The average accuracy results of NB and two combinations (NB, CombSW, CombV) on our 31 word subset are 86%, 73.1%, 72.1% respectively which are lower than our results; see Table 4.

These results suggest that our technique is fairly successful and promising and thus more research work should be exerted to carry out and further improve the performance of this technique.

Word	**Baseline (mfs)**	**EV1**	**EV2**	**EV3**
adjustment	0.67	0.99	0.96	0.93
blood_ pressure	0.54	0.98	0.80	0.83
cold	0.91	0.94	0.92	0.95
condition	0.98	0.95	0.95	0.95
culture	0.89	0.87	0.96	0.94
degree	0.97	0.93	0.93	0.93
evaluation	0.50	0.98	0.82	0.85
extraction	0.94	0.94	0.93	0.94
failure	0.86	0.83	0.83	0.83
fat	0.97	0.93	0.93	0.93
ganglion	0.93	0.93	0.91	0.93
glucose	0.91	0.90	0.90	0.93
growth	0.63	0.92	1.00	0.96
Immune suppression	0.59	0.98	0.88	0.87
implantation	0.83	0.91	0.96	0.87
japanese	0.92	0.92	0.97	0.92
lead	0.93	0.84	0.84	0.84
man	0.63	0.98	0.90	0.92
mosaic	0.54	0.99	0.77	0.87
nutrition	0.51	0.94	0.70	0.88
pathology	0.86	0.79	0.96	0.92
radiation	0.62	0.83	0.93	0.89
reduction	0.82	0.63	0.63	0.63
repair	0.76	0.92	0.91	0.96
sex	0.80	0.94	0.97	0.88
support	0.80	0.67	0.67	0.67
surgery	0.98	0.95	0.95	0.95
ultrasound	0.84	0.93	0.93	0.91
variation	0.80	0.86	0.94	0.89
weight	0.55	0.83	0.57	0.85
white	0.54	1.00	0.69	0.77
Mean Accuracy	**0.775**	**0.903**	**0.87**	**0.88**

Table 3: Detailed accuracy results of three evaluations EV1, EV2, and EV3.

References

1. M. Stevenson, Y. Guo, R. Gaizauskas, and D. Martinez. Knowledge Sources for Word Sense Disambiguation of Biomedical Text. BioNLP 2008, pp.80-87.
2. E. Agirre, A. Soroa and M. Stevenson. Graph-based Word Sense Disambiguation of biomedical documents. Bioinformatics, Vol. 26 no. 22, 2010, pp. 2889–2896
3. A. J. Jimeno-Yepes and A.R. Aronson. Knowledge-based biomedical word sense disambiguation: comparison of approaches. BMC Bioinformatics 2010, 11:569.
4. P. Chen and H. Al-Mubaid. "Context-based Term Disambiguation in Biomedical Literature". Proceedings of FLAIRS-2006. Orlando, Fla, USA, 2006.
5. H. Al-Mubaid and P. Chen. "Biomedical Term Disambiguation: An Application to Gene-Protein Name Disambiguation". Proc. of ITNG-2006, USA, 2006, pp. 606-612.
6. Stevenson, M. et al. Disambiguation of biomedical text using a variety of knowledge sources. BMC Bioinformatics, 2008, 9 (Suppl. 11), S7.
7. L. Humphreys, D. Lindberg, H. Schoolman, and G. Barnett. The Unified Medical Language System: Informatics Research Collaboration. J. of the Ameri Medical Informatics Association, 1(5), 1998.
8. G.K. Savova, et al. Word sense disambiguation across two domains: biomedical literature and clinical notes. J. Biomed. Informatics, vol. 41, 2008, pp.1088–1100.
9. S. Humphrey et al. Word Sense Disambiguation by selecting the best semantic type based on Journal Descriptor Indexing. J. American Soc. of Informatics, vol. 57, 2006, pp. 96–113.
10. G. Forman. An Extensive Empirical study of feature selection metrics for text classification. JMLR, 2003.
11. L. Galavotti, F. Sebastiani, M. Simi. Experiments on the use of feature selection and negative evidence in automated text categorization. The 4th European Conf. on Research and Advanced Technology for Digital Libraries. 2000.
12. Y. Yang, J.P. Pedersen . A comparative study on feature selection in text categorization. The 4th Intl Conf. on Machine Learning, 1997
13. Z. Zheng, R. Srihari. Optimally combining positive and negative feature for text categorization, ICML'2003 Workshop on Learning from Imbalanced Data Sets, 2003.
14. T. Joachims. Text categorization with support vector machines: learning with many relevant features. 10th European Conference on Machine Learning, 1998.
15. M. Weeber, J. Mork, and A. Aronson. Developing a test collection for biomedical word sense disambiguation. Proceedings of AMIA Symposium American Medical Informatics Association; 2001.

16. M.F. Porter. An algorithm for suffix stripping. Program,14:130–137, 1980.

17. J-W Son and S-B Park. Learning Word Sense Disambiguation in Biomedical Text with Difference Between Training and Test Distributions. DTMBIO'09, November, 2009.

18. H. Xu, M. Markatou, R. Dimova, H. Liu and C. Friedman. Machine learning and word sense disambiguation in the biomedical domain: design and evaluation issues. BMC Bioinformatics, 2006, 7:334.

Word	**Baseline (mfs)**	**Previous Results**									**Our method (EV1)**
		Stevenson et al. (2008) [1]			Agirre et al. (2010) [2]			Jimeno-Yepes et al. (2010) [3]			
		Joshi - 2005	**McInnes 2007**	**Stevenson-2008**	**Single**	**Subset**	**Full**	**NB**	**CombSW**	**CombV**	
adjustment	67	71	70	74		33.3	35.5	76.3	69	53.9	99
blood pressure	54	53	46	46	53.0	50	48	57.0	38	44	98
cold	91	90	89	88	32.6	26.3	28.4	92.6	39	79	94
condition	98	-	89	89	95.7	39.1	48.9	97.8	78	69	95
culture	89	-	94	95		33	77	93.0	100	54	87
degree	97	89	79	95		95.4	93.8	96.9	88	82	93
evaluation	50	69	73	81	59	54	50	78.0	52	50	98
extraction	94	84	86	85		23	27.6	94.3	98	86	94
failure	86	-	73	67		27.6	72.4	86.2	86	100	83
fat	97	84	77	84	56.2	63	95.9	97.3	91	84	93
ganglion	93	-	94	96	66	77	64	95.0	88	86	93
glucose	91	-	90	91	91	91	90	91.0	78	39	90
growth	63	71	69	68	37	37	37	73.0	55	66	92
Immune suppression	59	80	75	80	64	59	62	79.0	60	65	98
implantation	83	94	92	93	75	84.7	84.7	98.0	94	97	91
japanese	92	77	76	75	70.9	70.9	64.6	92.4	63	94	92
lead	93	89	90	94	93.1	93.1	93.1	93.1	83	86	84
man	63	89	80	90	61.5	34.8	44.6	87.0	65	42	98
mosaic	54	87	75	87		60.8	66	82.5	84	72	99
nutrition	51	52	49	54		33.7	32.6	55.1	45	43	94
pathology	86	85	84	85		34.3	28.3	85.9	76	83	79
radiation	62	82	81	84	58.2	53.1	53.1	83.7	76	76	82
reduction	82	91	92	89	36.4	54.5	54.5	81.8	100	82	63
repair	76	87	93	88	63.2	72.1	76.5	95.6	87	88	92
sex	80	88	87	87	84	85	85	84.0	60	53	94
support	80	-	91	89	80	80	80	80.0	100	90	67
surgery	98	-	94	97	95.9	97	97	98.0	43	96	95
ultrasound	84	92	85	90	84	84	83	85.0	81	83	93
variation	80	-	91	95	85	80	75	91.0	65	86	86
weight	55	83	79	81	56.6	56.6	56.6	84.9	66	68	83
white	54	79	74	76	68.9	67.8	63.3	81.1	57	58	100
Average	**77.5**	**81.1**	**81.2**	**83.6**	**68.8**	**59.7**	**63.5**	**86.0**	**73.1**	**72.7**	**90.3**

Table 4: Comparison of our results with the best reported results from recent reported techniques.