Hyperdimensional Computing Approach to Word Sense Disambiguation

Bjoern-Toby Berster, MS[1], J Caleb Goodwin, MSEE, MSBMI[1], Trevor Cohen, MBChB, PhD[1, 2]
[1]University of Texas Health Science Center at Houston : School of Biomedical Informatics
[2]National Center for Cognitive Informatics and Decision Making in Healthcare

Abstract

Coping with the ambiguous meanings of words has long been a hurdle for information retrieval and natural language processing systems. This paper presents a new word sense disambiguation approach using high-dimensional binary vectors, which encode meanings of words based on the different contexts in which they occur. In our approach, a randomly constructed vector is assigned to each ambiguous term, and another to each sense of this term. In the context of a sense-annotated training set, a reversible vector transformation is used to combine these vectors, such that both the term and the sense assigned to a context in which the term occurs are encoded into vectors representing the surrounding terms in this context. When a new context is encountered, the information required to disambiguate this term is extracted from the trained semantic vectors for the terms in this context by reversing the vector transformation to recover the correct sense of the term. On repeated experiments using ten-fold cross-validation and a standard test set, we obtained results comparable to the best obtained in previous studies. These results demonstrate the potential of our methodology, and suggest directions for future research.

Introduction

A fundamental problem of Information Retrieval (IR) systems and Natural Language Processing (NLP) is determining the correct sense of a word in a given context. A classical example is the word "bank" which can take on different meanings in the context of the word "river" referring to the land on the edge of a river or "money" as in a financial institution. In IR, WSD can result in a mismatch between the document and the query, if either is using a different meaning of a given term [1]. In NLP, the problem arises in the area of text classification [2]. WSD approaches have led to performance improvements for both NLP and IR applications [1, 3].

This paper presents an approach, in which high-dimensional vector representations are used to model the meaning of words and their surrounding context. This approach is closely related to the family of models which include Latent Semantic Analysis (LSA) [4], Random Indexing (RI) [5], and Reflective Random Indexing (RRI) [6]. These approaches stem from the distributional hypothesis, which asserts that the meaning of a word can be defined based on the contexts in which it occurs [7], and as such can be categorized as methods of distributional semantics [8]. However, in contrast to the aforementioned approaches, this work utilizes Pennti Kanerva's Binary Spatter Code (BSC) [9], which enables the encoding of typed relations between concepts, and operations such as processing of analogies [10, 11].

The BSC is part of a family of representational approaches collectively known as Vector Symbolic Architectures (VSA) [12], which are able to accomplish processes usually associated with symbolic systems using reversible vector transformations. For example, reversible transformations of binary vectors have been used to encode typed relations into a distributional space in order to support analogical reasoning [10, 11, 13]. Additionally, the VSA models maintain desirable properties of connectionist systems. That is, knowledge is encoded in a VSA as a distributed representation that maintains the desired properties of being resistant to noise as well as naturally supporting distance metrics between concepts while avoiding biologically implausible and expensive back propagation [12]. The mathematical operators utilized by VSAs are generally applicable, and can be used to augment distributional models with additional knowledge. In this paper, we use these operators to encode the sense of an ambiguous term, and hypothesize that a concept space augmented in this manner can be used to improve WSD performance.

The remainder of the paper is organized as follows. We review other WSD approaches, ranging from statistical models to knowledge-based approaches, and consider their results for later comparison to our own. We also introduce some background material about our chosen approach, to set the stage for the description of our methods that follows. The methods section introduces our methods as well as the utilized data set, following which we present our results and compare these to results obtained using other approaches. Lastly we discuss our results and conclude with future directions for the line of research developed in the current study.

Background and related work

Numerous approaches have been applied to the task of WSD. Space constraints preclude an exhaustive review of all of these approaches, but we refer the interested reader to the detailed accounts to be found in [13-15]. The approaches to WSD can largely, but not exclusively, be categorized into machine learning [14, 15], knowledge-based approaches [15, 16], statistical or distributional semantic approaches [17], and graph-based approaches [18].

Machine learning methods utilize a combination of machine learning classifiers, often a form of naïve Bayes, and feature sets, such as sliding window sizes, collocations, and part-of-speech. Knowledge-based approaches introduce outside knowledge to machine learning methods. Statistical or distributional semantic approaches incorporate the distributional theory, referenced in the introduction of our paper, such as LSA, RI or RRI. The graph-based methods, utilize graph-based algorithm, such as PageRank, to disambiguate words in specific contexts, and most of these methodologies are applied in an unsupervised context.

For the remainder of the review, we focus on methods that utilized the same data set we have used and can therefore provide benchmarks for our experiments. Three specific approaches were of interest to us, of which two can be categorized as supervised machine learning methodologies that utilize features present in the data set only [2, 19] and one can be categorized as a supervised machine learning that draws upon additional knowledge sources [20].

Liu et al. [2] compared various supervised learning algorithms, such as naïve Bayes, traditional decision list learning, mixed supervised learning, and their own implementation of decision list learning. The effects of different combinations of feature representations, such as varying window size around the ambiguous word, distance to the main target word, left or right orientation of context word in relation to the ambiguous word, collocations, and unigrams were also evaluated. This analysis was not restricted to the data set used in this study, but included also additional data sets, involving abbreviations and general English words. Upon evaluation, their mixed supervised learning implementation outperformed the other machine learning methods. In particular, the naïve Bayes classifiers fared poorly. For each of the examined ambiguous word, the authors reported the accuracy of the best performing combination of feature and classifier, and across all of these approaches accuracy was reported at 85.58%.

Joshi et al. [19] applied features to the biomedical domain that have been used for general English word sense disambiguation, using a range of classifiers including Support Vector Machines. The features Joshi et al. [19] employed were based on unigrams and bigrams of words neighboring the ambiguous word. The accuracy of 85.14% reported by Joshi stems from the selection of the highest result among all investigated combinations of features and classifiers for each ambiguous word in a subset of the total test set that they defined.

Leroy and Rindflesch [20] hypothesized that the combination of machine learning methodologies and additional biomedical knowledge, in the form of UMLS, would improve performance in the WSD task. Different features were analyzed but their best performing feature set with a Naïve Bayes classifier considered the grammatical role of the ambiguous word, it's part of speech, and the semantic types of the words in the context sentence. Upon evaluation, accuracy was measured at 67.80%.

Mathematical Structure and Methods

As previously stated, our approach stems from Kanerva's work on hyperdimensional computing [9, 10, 13, 21], which uses as a representational unit high-dimensional binary vectors, where dimensionality is on the order of 10,000 dimensions or more. While there are many ways to generate such representations, we base our approach upon the RI paradigm using the terminology as specified in [6], in which a *semantic vector* for a term is generated by superposing randomly generated *elemental vectors* that represent the contexts this term has occurred in. Elemental vectors are constructed by randomly assigning ones and zeros to the dimensions of a high-dimensional binary vector with equal probability, such that there is a high probability of these vectors being approximately orthogonal to one another [21], with orthogonality in the binary vector space being defined as the Hamming distance of half of the dimensionality of the space [21]. Semantic vectors for terms are constructed as superpositions of the elemental vectors for the terms they occur with. Throughout the rest of this work we will refer to elemental vectors of term X as $E(X)$ and semantic vectors of the same term as $S(X)$. In addition to elemental vectors for terms, we introduce elemental vectors for the senses of an ambiguous term, such that $E(X_i)$ represents the elemental vector for the ith sense of term X.

It is important to note that the BSP high-dimensional binary vectors are different in nature from term co-occurrence vectors used in many supervised learning approaches. A common representation in NLP or WSD algorithms is for each element of the vector to represent the number of co-occurrences between one term and another, or to represent the presence or absence (i.e. binary representation) of this co-occurrence. In contrast, the vectors used in the BSC are fully

distributed representations in the sense that no individual dimension is meaningfully interpretable. Information regarding each co-occurring term exists as a pattern distributed across an entire vector. This confers robustness to the model making it tolerant to error, and enabling the recognition of encoded elemental vectors despite their being distorted during the training process. According to Kanerva, binary vectors can be considered as the "basic unit on which to compute" with motivation of BSC research being the determination of "how to load specific data into these vectors and how to extract data from them" [10]. The data loaded into the binary vectors in this case are the context in which the different senses of a word occurs. The purpose of the training is to encode these data (in this case context) in the high-dimensional space.

VSA have two primary operations, namely binding and bundling. Binding is a multiplication-like operator through which two vectors are aggregated to form a third vector C that is different from either of its component vectors A and B. Throughout this paper we will refer to this operation with the symbol "⊗" and use the symbol "⊘" for the inverse of binding. The implementation of binding differs from model to model, but always has the following features: (1) it produces a product of the same dimensionality as the component vectors (unlike the inner or outer product of linear algebra); and (2) it is invertible, such that the information encoded by binding can be recovered. That is to say, the vector A can be applied to the vector product C (which equals A ⊗ B) to recover B. Conversely, B can be applied to C to recover A. Consequently, we are able to retrieve information, which was encoded in the binding process, by applying the inverse function, ⊘. In the BSC, both binding and its inverse are accomplished using bitwise exclusive or (XOR), which is its own inverse, as illustrated in Table 1:

A	B	$C = A \otimes B$	$C \oslash B$	$C \oslash A$
10111	01001	11110	10111	01001

Table 1: XOR as a reversible binding operator for binary vectors

Bundling on the other hand resembles addition, and provides the means to accomplish superposition of vectors. Like binding, the implementation of bundling differs across VSA models. In some implementations, this occurs through vector addition, but in the case of the binary implementation used in the current research, it occurs by maintaining a "voting record" of the number of ones and zeros added to each dimension of the bundled vector, and assigning a one to a given dimension of the resulting vector if more ones than zeros are encountered, and vice versa, with ties broken at random. We will use the symbol "+" for bundling. Furthermore, we will utilize the symbol combination of "+=", commonly used in computer science, to denote bundling the vector on the left hand side with the vector on the right side of the equal sign and taking the outcome as the new resulting vector. For example the operation S(A) += E(X) could be rewritten as S(A) = S(A) + E(X) (i.e. add the elemental vector representing term X to the semantic vector representing term A) [6].

In the research presented in this paper, we apply the VSA operators to encode different senses of ambiguous terms into the semantic vector representations of the terms that occur with them. Consider the case of the ambiguous term "cold", occurring in the context of the phrase "a cold is a childhood infection", annotated with the "common_cold" sense of the term. One of the steps during the process of training would be to bind this term to its relevant sense, and bundle this bound product into the semantic vector for the term "infection", as follows:

$$S(\text{infection}) += E(\text{cold}) \otimes E(\text{common\_cold})$$

If the term "cold" is encountered in a context that includes the term "infection", S(infection) forms a component of the context vector for this context. Consequently, when E(cold) is released from this context vector, an approximation of the elemental vector representing the correct sense as recovered:

$$\begin{aligned} & S(\text{context vector}) \oslash E(\text{cold}) \\ & \approx S(\text{infection}) \oslash E(\text{cold}) \\ & \approx E(\text{cold}) \otimes E(\text{common cold}) \oslash E(\text{cold}) \\ & \approx E(\text{common cold}) \end{aligned}$$

If the term "infection" has occurred with other terms, or senses, in other contexts the recovered vector will be a noisy approximation of E(cold), however the dimensionality of the space, and the near orthogonal nature of the elemental

vectors it contains ensure that this approximation is likely to be considerably closer to *E*(cold) than it is to any other elemental vector in the space.

Methods

BSC-WSD algorithm implementation and design

We developed the BSC-WSD algorithm based on the BSC implementation that is available as a component of the Semantic Vectors open source package for distributional semantics research [22-24]. The Semantic Vectors package provides a set of classes and methods to facilitate the generation of random elemental vectors, training of semantic vectors from the elemental vectors, and mathematical operations such as the "bind" and "release" operators [25]. In addition, the Semantic Vectors package extends the highly scalable and widely adopted Apache Lucene API [26] for storing and extracting distributional statistics from the corpus.

The training begins by creating elemental vectors, which are high dimensional binary vectors that have an equal number of zeros and ones assigned at random to the elements of the vector. To maintain consistency across experiments, we used a deterministic variant of the RI approach to generate elemental vectors, such that the same elemental vector was used for each ambiguous term and sense across experiments [27]. Elemental vectors are used to create trained semantic vectors for the terms in the corpus based on the occurrence of ambiguous terms and the context in which these terms occur. The weight of each term is determined using entropy weighting, which increases the relative contribution of terms that occur focally, and are therefore likely to be more informative.

$$entropy(\text{term}_i) = 1 + \sum \frac{P_{ij} log_2(P_{ij})}{log_2(n)}$$

where P_{ij} = (occurrences of term *i* in document *j*) / (global frequency of term *i*)

n = the number of documents in the corpus

This weighting is used to determine the number of "votes" assigned to the contribution of each term vector when tallying the number of 1's and 0's in each dimension of the superposed product.

The creation of the semantic term vectors from the elemental term vectors proceeds as follows:

1. Create elemental vectors by randomly assigning 1's and 0's to the elements of vectors representing:
    a. Each sense, which is defined by a UMLS concept, of an ambiguous term *E*(s).
    b. Each ambiguous term in the data set *E*(ambiguous term).
2. For each word in the context (ST = sentence and title or AT = abstract and title), we used the binding operation for the elemental vector of the ambiguous word in the sentence with the assigned sense of the target word and bundled this to the semantic vector of the word in the context. Represented symbolically:

$$S(\text{context term}) += E(\text{ambiguous term}) \otimes E(\text{relevant sense})$$

This process was repeated across all of the training contexts, such that the semantic vector for a term ultimately encoded many different ambiguous terms and word senses.

Once the semantic vectors are created for the terms, these trained vectors can be used to determine the appropriate sense of a word from a sentence or abstract. Given a sentence or abstract, a context vector is created by superposing (bundling) the trained semantic terms vectors for each term in the sentence as follows:

$$S(\text{context}) += \sum_{k=1}^{n} S(\text{context term}_i)$$

Binary local weighting was applied, which is to say that each term in the context is counted once only. This prevents terms that occur frequently in a given context from dominating the context vector representation. To identify the

appropriate sense of an ambiguous term, the release or inverse binding operator is applied to the context vector *S*(context) and the elemental vector *E*(ambiguous term) as follows:

$$S(\text{context}) \oslash E(\text{ambiguous term}) \approx E(\text{relevant sense})$$

To attempt to recover the appropriate sense from this vector product, we compare it to the elemental vectors for all of the encoded senses using one minus the normalized Hamming distance. We included in this search all of the encoded senses across all terms, without restricting search to those senses relevant to a particular ambiguous term, as we would anticipate the decoding process mapping to a region of hyperdimensional space that contains the senses relevant to this term only.

Evaluation and experimental configuration

The National Library of Medicine (NLM) provides a dataset for researchers attempting to improve the current state of the art for WSD [28]. We will henceforth refer to this dataset as the NLM-WSD dataset. This data set is available from **[28]** and contains 50 English terms, which are commonly thought of as ambiguous. Such terms include "cold", "growth" and "lead" and each of these is mapped to multiple UMLS concepts representing the various senses of the term. The data set contains 100 instances for each word, which were disambiguated by domain experts. These experts tagged the corpus with what they believe to be the proper sense, or with "None" if none of the available options were perceived to have described the correct sense. Researchers at the NLM then reviewed the tagged corpus and we used this reviewed data set for our experiments. The format of the data set includes the original MEDLINE abstract, the PubMed ID of the article, the assigned sense, the title, and the sentence that includes the ambiguous word. The 100 abstracts for each term were randomly selected from a total a set of 409,337 MEDLINE citations.

We compare our results to three other studies, which used the NLM-WSD data set. Each of these prior works used a custom subset of the NLM-WSD data set. Leroy and Rindflesch **[20]** established a subset containing only 15 words, with the criterion that the majority sense was correct less than 65% of the instances. For example, the term "variation" was excluded because the sense M2 was assigned in 80% of all cases and exceeded the threshold of 65%. The subset created by Liu et al. **[29]** contains 22 terms, excluding 12 ambiguous terms, which are considered problematic by Weeber et al. due to the fact that little to no agreement among raters was observable, causing many ties and majorities of one **[30]**. Furthermore, 16 additional terms were excluded because the majority sense occurred more than 90% of instances. Joshi et al. **[19]** used every word for their initial training of the model and excluded the words for which none of their classifiers could achieve at least five percentage point accuracy improvement over the majority sense. Hence the final subset included 30 words. We will refer to these subsets as Leroy, Liu, and Joshi respectively, in our results section.

In addition to replicating the experimental protocols of Leroy and Rindflesch, Liu et al., and Joshi et al., we performed several experiments on the entire data set. We tested the following definitions of context: (i) the title and the sentence containing the ambiguous term as a context (TS); and (ii) title and the abstract as the context for the ambiguous term (TA). For the majority of these configurations, we explored the impact on performance of using vectors of different dimensionality.

In these experiments, we utilized the stop word list generated by Salton and Buckley for the experimental SMART information retrieval system [31]. In addition a threshold was used to eliminate terms that had a frequency of occurrence less than 2 or greater than 1,500. Given that this approach is a supervised machine learning method, we split our data set into training and test sets. For all of the experiments we used ten-fold cross validation. Each run produced different measures of accuracy and the final accuracy is then reported as the average over all ten runs.

For a term *t*, accuracy is defined as . $\frac{\text{correctly labeled instances of } t}{\text{number of instances of } t}$

Results

Tables 2, 3, and 4 present the results of our experiments. The label BSC stands for Binary Spatter Code, and is the approach explored in this paper. The label Liu, Leroy, and Joshi contain the best results from the algorithms contained in [29] , [20] and [19]. These approaches are discussed in detail in the methods section of our work. The label "S" refers to sentence and title and the label "A" refers to abstract and title. For example, BSC-S is the binary vector WSD algorithm using an individual sentence and title as context and BSC-A is the binary vector WSD algorithm using the abstract and title as context. The numbers for each version of the binary vector WSD algorithm are 8, 16, and 65 represent the dimensionality of the binary vector which are 8,192, 16,384, and 65,536 bits respectively. The labels "L" and "R" refer to experiments using the protocol of Liu et al. ("L") and Leroy and Rindflesch ("R"), which are described in the Methods section. In interpreting the results, one should keep in mind that Liu et al. reported only the highest yielded result for each

ambiguous word with a certain classifier and feature set. The classifiers and feature sets vary across their report of accuracy. For example, Liu et al. reported for the term "cold" an accuracy of 90.9, which was the highest among all experiments, but the feature set used to obtain this result may not be the same one used to obtain the most accurate result on one of the other terms.

Table 2: Comparison of results to other disambiguation studies and reporting of own results.

Ambiguous Term	BSC-S16-L	BSC-S65-L	BSC-A16-L	BSC-S16-R	BSC-S65-R	BSC-A16-R	BSC-S8	BSC-S16	BSC-S65	BSC-S262	BSC-A16	BSC-A65
adjustment				55.91	58.06	55.91	68.82	69.89	70.97	70.97	67.74	67.74
blood pressure				60	59	52	58.00	56.00	57.00	58.00	50.00	51.00
cold	90.53	90.53	90.53				93.68	91.58	92.63	92.63	90.53	90.53
condition							96.74	96.74	96.74	96.74	97.83	97.83
culture							96.00	97.00	97.00	97.00	90.00	89.00
degree	95.38	96.92	96.92	73.85	81.54	61.54	93.85	93.85	96.92	96.92	96.92	96.92
depression	96.47	96.47	100				96.47	95.29	96.47	96.47	100.0	100.00
determination							97.47	98.73	98.73	98.73	100.0	100.00
discharge	98.67	98.67	98.67				98.67	98.67	98.67	98.67	98.67	98.67
energy							94.00	97.00	99.00	99.00	99.00	99.00
evaluation				54.55	57.58	64	61.00	59.00	66.00	66.00	52.00	60.00
extraction	92.94	94.12	94.25				94.19	94.19	94.19	94.19	94.25	95.40
failure							68.97	62.07	72.41	82.76	86.21	62.07
fat	94.52	94.52	97.26				94.52	93.15	94.52	94.52	97.26	97.26
fit							77.78	88.89	94.44	94.44	100.0	94.44
fluid							99.00	100.00	100.00	100.00	100.0	100.00
frequency							95.74	100.00	100.00	100.00	100.0	100.00
ganglion							94.00	94.00	94.00	94.00	93.00	93.00
glucose							91.00	90.00	91.00	91.00	91.00	91.00
growth	69.00	72	68	61.62	62.63	62	71.00	71.00	70.00	69.00	70.00	71.00
immuno-suppression				66	66	79	68.00	68.00	68.00	71.00	68.00	73.00
implantation	94.90	93.88	91.84				93.88	93.88	93.88	94.90	87.76	96.94
inhibition							97.98	97.98	97.98	97.98	98.99	98.99
japanese	86.08	87.34	92.41				86.08	86.08	88.61	88.61	92.41	92.41
lead	71.43	71.43	93.1				71.43	64.29	71.43	71.43	93.10	93.10
man	84.78	83.7	81.52	80.43	80.43	69.57	80.43	83.70	86.96	88.04	73.91	84.78
mole	98.81	98.81	98.81				98.81	98.81	98.81	98.81	98.81	98.81
mosaic	84.54	85.57	82.47	84.54	83.51	85.57	85.57	85.57	85.57	85.57	89.69	84.54
nutrition	52.81	52.81	49.44	53.93	52.81	44.94	52.81	53.93	52.81	52.81	50.56	48.31
pathology	85.86	86.87	86.87				85.86	85.86	87.88	88.89	85.86	85.86
pressure							93.75	95.83	96.88	96.88	2.08	4.17
radiation				72.45	77.55	77.55	77.55	83.67	85.71	85.71	76.53	77.55
reduction	63.64	72.23	81.82				63.64	63.64	72.73	81.82	9.09	63.64
repair	80.88	82.35	82.35	82.35	86.76	88.24	79.41	80.88	79.41	82.35	77.94	83.82
scale	98.46	98.46	100	89.23	92.31	80	100.0	98.46	100.00	100.00	100.0	100.00
secretion							99.00	99.00	99.00	99.00	99.00	99.00
sensitivity				90.2	92.16	86.27	96.08	96.08	96.08	96.08	96.08	96.08
sex	83.84	83.84	80				85.00	83.00	83.00	82.00	81.00	81.00
single							95.00	95.00	99.00	99.00	99.00	99.00
strains							97.85	97.85	97.85	97.85	98.92	98.92
support							50.00	50.00	50.00	60.00	20.00	70.00
surgery							98.00	98.00	98.00	98.00	98.00	98.00
transient							98.00	98.00	98.00	99.00	99.00	99.00
transport							97.87	98.94	98.94	98.94	98.94	98.94
ultrasound	81.82	81.82	84				82.00	81.00	83.00	82.00	84.00	84.00
variation							90.00	89.00	90.00	88.00	85.00	81.00
weight	84.91	86.79	77.36	64.15	67.92	41.51	66.04	81.13	79.25	88.68	71.70	83.02
white	76.67	76.67	73.33	68.89	68.89	54.44	74.44	76.67	74.44	75.56	75.56	66.67
	84.86	85.72	86.41	70.54	72.48	66.84	85.53	86.07	87.37	88.33	83.03	85.32

In Table 2, we show our results for each of the 50 terms included in the data set, and for the subsets of these terms evaluate by Liu (L) and Leroy and Rindflesch (R). The results in Table 3 contain the scores for different methods for these terms, as well as the results of BSC-S-16 and BSC-A-65, which were the highest performing configurations of the BSC-WSD algorithm. The results in Table 4 are the union of the words used in the analysis of Liu et al., Leroy and

Rindflesch, and Joshi et al. The results in Table 4 present the performance of different configurations of BSC and the results of comparative solutions on the subsets. The highest performing system across all of the configurations was BSC-S-65.

Table 3: Comparison of results. Best performance is in boldface.

Ambiguous Word	Liu	Leroy	Joshi S	BSC-S-65	BSC-A-65	Best BSC
adjustment		62.0	**71.0**	70.97	67.74	70.97
blood_pressure		56.0	**53.0**	57.00	52.00	60
cold	90.9		90.0	**92.63**	90.53	93.68
degree	**98.2**	70	89.0	96.92	96.92	96.92
depression	88.8		86.0	96.47	**100.00**	100.00
discharge	90.8		95.0	**98.67**	**98.67**	98.67
evaluation		57	**69.0**	66.00	60.00	66.00
extraction	89.7		84.0	94.19	**95.40**	95.40
fat	85.9		84.0	94.52	**97.26**	97.26
growth	72.2	63	**71.0**	70.00	65.00	72.00
immunosuppression		67	**80.0**	68.00	72.00	79.00
implantation	90		**94.0**	93.88	87.76	96.94
japanese	79.8		77.0	88.61	**92.41**	92.41
lead	91		89.0	71.43	**93.10**	93.10
man	91	80	**89.0**	86.96	76.09	88.04
mole	91.1		95.0	**98.81**	**98.81**	98.81
mosaic	**87.8**	69	87.0	85.57	85.57	89.69
nutrition	58.1	53	**52.0**	52.81	50.56	53.93
pathology	**88.2**		85.0	87.88	85.86	88.89
radiation		72	82.0	**85.71**	79.59	85.71
reduction	91		**91.0**	72.73	63.64	81.82
repair	76.1	81	**87.0**	79.41	77.94	88.24
scale	90.9	84	88.0	**100.00**	**100.00**	100.00
sensitivity		70	88.0	**96.08**	**96.08**	96.08
sex	**89.9**		92.0	83.00	81.00	85.00
strains			83.0	97.85	**98.92**	98.92
ultrasound	87.8		**79.0**	83.00	84.00	84.00
weight	78	71		79.25	**84.91**	88.68
white	75.6	62		74.44	68.89	76.67

Table 4: Results based on subsets

	Liu subset	Leroy subset	Joshi subset
Liu	85.58	*NA*	85.2
Leroy	*NA*	67.80	77.38
Joshi	86.82	80.47	84.15
BSC-S-65	85.51	77.94	83.54
BSC-A-65	85.20	75.55	82.78
BSC-S-16-R	*NA*	70.54	*NA*
BSC-A-16-R	*NA*	66.84	*NA*
BSC-S-65-R	*NA*	72.48	*NA*
BSC-S-16-L	84.86	*NA*	*NA*
BSC-A-16-L	86.41	*NA*	*NA*
BSC-S-65-L	85.72	*NA*	*NA*
Best BSC	**89.10**	**81.00**	**87.09**

Table 5 presents a summary of the impact on performance based on the number of dimensions used and the type of context utilized for disambiguation. A reasonable hypothesis might be that the use of the sentence containing the ambiguous term would improve performance, while the use of the entire abstract would cause deterioration, due to the introduction of noise. The results in Table 5 show that using only the sentence containing an ambiguous word resulted in a modest improvement in performance. Conversely, reducing the dimensionality of the binary vectors led to a modest decrease in performance. This can be explained by an increase in random overlap between elemental vectors at lower dimensionality, and can be anticipated to some extent by determining the capacity of elemental vectors of a particular dimensionality empirically [27].

Table 5: Impact of dimensionality on performance. S = sentence. A = abstract.

	8,192 dimensions	16,384 dimensions	65,536 dimensions
BV-S	85.53	86.07	87.37
BV-A	83.03	85.32	85.23

Discussion and future work

This study presented a new approach to WSD leveraging high-dimensional binary vectors. We found that the BSC-WSD algorithm compared favorably to established approaches and resulted in performance improvements across the data sets investigated. Our approach led to significant performance increases over baseline measures. In addition to performance increases, another strength of this approach is scalability. The process of training our vectors and then testing with dimensions of 8,192 and 16,384 took on average between 5 and 15 minutes in the cases of sentences on a laptop with 8GB of RAM and a 2.0 GHz processor. In addition, the accuracy of our system is particularly noteworthy given that it has only a small probability of obtaining the correct answer by chance. For example, in a traditional machine learning approach a given item could have the label *A*, *B*, or *C*. The probability of guessing the correct label in this case is $\frac{1}{3}$. In our approach, the probability of guessing a correct label for all terms is $\frac{1}{\text{number of senses}}$, (approximately $\frac{1}{100}$).

The primary weakness of this study is the size of the test set used. However, this is a weakness of all WSD studies using the NLM-WSD data set. One limitation that others have mentioned as well is the small size of the corpus. Although we used 10 fold cross validation evaluation method, we do not know how our approach, or other approaches, will perform in other contexts.

An interesting finding of these experiments is that considering a larger context does not decrease the performance substantially. The abstract had on average a context size of 220 words, while the sentence had only 26 words on average, and the decrease in performance ranged between 1% and 2% the above configurations. This behavior can be explained by the presence of terms in this larger context that are not relevant to the sense of the ambiguous term. There are several areas for future research leveraging this approach. A known method for improving accuracy of distributional models is using sliding windows to encode word order as context. It is reasonable to assume that this approach would result in improved performance and is an area that we plan to explore in the future. In addition, we found that context based on abstract and title versus sentence and title contained different information. This suggests it may be possible to combine the results of the abstract and sentence-based models by selecting the sense with strongest association across both models, or weighting the results obtained with each model using an ensemble approach.

This work has several contributions. Firstly, this is the first application of VSA models and the BSC for biomedical WSD. This approach is fundamentally different from established methods of supervised training, in that we do not train a classifier for each sense of each term in the training set. Rather, the information required to disambiguate an ambiguous term is distributed across the semantic vector representations of every other term in the corpus. When this term is encountered in a new context, these vectors are superposed and the elemental vector for the correct sense is reconstituted. Secondly, we showed that the BSC model has comparative performance with the state-of-the-art WSD techniques. Finally, from a theoretical standpoint, the use of the BSP to perform WSD is important given its theoretical cognitive underpinnings. In addition to improved WSD systems, this may provide a model for the interpretation of language during human cognition.

Conclusion

In this work we presented a new approach to WSD using binary vectors to implement a VSA. The performance was close or slightly better to other reported instances, and performing further experiments might reveal further improvements compared to other methods. Promising directions for future research include the evaluation of other approaches to modeling context, the application of this approach to related problems, such as the disambiguation of acronyms, and the further exploration of distributed representations of this nature as a means to perform classification tasks in general.

Acknowledgements

This research was supported in part by the US National Library of Medicine grant (R21LM010826-01), Encoding Semantic Knowledge in Vector Space for Biomedical Information Retrieval. We would also like to acknowledge Dominic Widdows, originator of the Semantic Vectors Package, who developed the mathematical notation used to describe the BSC operations. In addition, this research was funded in part by a training fellowship from the Keck Center of the Gulf Coast Consortia, on the Training Program in Biomedical Informatics, National Library of Medicine (NLM) T15LM007093, PI - G. Anthony Gorry.

REFERENCES

[1] C. Strokoe, M. J. Oakes, and J. I. Taft, "Word sense disambiguation in information retrieval revisited," presented at the In Proceedings of the 26th Annual Internation ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, Onto., Canada, 2003.

[2] H. Liu, "Automatic Resolution of Ambiguous Terms Based on Machine Learning and Conceptual Relations in the UMLS.," *Journal of the American Medlical Informatics Association,* vol. 9, pp. 621–636, 2002.

[3] A. R. Aronson, "Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program," presented at the Proceedings of the AMIA Symposium, 17. American Medical Informatics Association., 2001.

[4] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science,* vol. 41, pp. 240–256, 1990.

[5] M. Sahlgren, " An introduction to random indexing. Methods and Applications of Semantic Indexing," presented at the Workshop at the 7th International Conference on Terminology and Knowledge Engineering, 2005.

[6] T. Cohen, R. Schvaneveldt, and D. Widdows, "Reflective Random Indexing and indirect inference: A scalable method for discovery of implicit connections," *Journal of Biomedical Informatics,* vol. 43, pp. 240-256, 2010.

[7] Z. Harris, "Distributional structure," *Word,* vol. 10, pp. 146-162, 1954.

[8] T. Cohen and D. Widdows, "Empirical distributional semantics: Methods and biomedical applications.," *Journal of Biomedical Informatics,* vol. 42, pp. 390-405, 2009.
[9] P. Kanerva, "Binary spatter-coding of ordered K-tuples.," presented at the Proc. ICANN'96, Bochum, Germany, 1996.
[10] P. Kanerva, "What We Mean When We Say "What's the Dollar of Mexico?": Prototypes and Mapping in Concept Space," presented at the 2010 AAAI Fall Symposium Series., 2010.
[11] T. Cohen, D. Widdows, R. Schvaneveldt, and T. C. Rindflesch, "Finding Schizophrenia's Prozac," presented at the Quantum Interaction: 5th International Symposium, Aberdeen, Uk, 2011.
[12] R. W. Gayler, "Vector Symbolic Architectures answer Jackendoff's challenges for cognitive neuroscience," presented at the ICCS/ASCS international conference on cognitive science, 2004.
[13] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive Computation,* vol. 1, pp. 139–159, 2009.
[14] G. Savova, T. Pedersen, A. Purandare, and A. Kulkarni, "Resolving ambiguities in biomedical text with unsupervised clustering approaches," University of Minnesota Supercomputing Institute2005.
[15] J. Antonio and M. Bridget, "Collocation analysis for UMLS knowledge-based word sense disambiguation," *BMC Bioinformatics,* vol. 12, 2011.
[16] S. M. Humphrey, W. J. Rogers, H. Kilicoglu, D. Demner-Fushman, and T. C. Rindflesch, "Word sense disambiguation by selecting the best semantic type based on Journal Descriptor Indexing: Preliminary experiment," *Journal of the American Society for Information Science and Technology,* vol. 57, pp. 96–113, 2005.
[17] D. M. Bikel, "A statistical model for parsing and word-sense disambiguation," presented at the Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora, Hong Kong, 2000.
[18] E. Agirre, A. Soroa, and M. Stevenson, "Graph-based Word Sense Disambiguation of biomedical documents," *Bioinformatics,* vol. 26, pp. 235-240, 2010.
[19] M. Joshi, T. Pedersen, and R. Maclin, "A comparative study of support vector machines applied to the supervised word sense disambiguation problem in the medical domain," presented at the Proceedings of the Second Indian International Conference on Artificial Intelligence, 2005.
[20] G. R. Leroy, T.C., "Using symbolic knowledge in the UMLS to disambiguate words in small datasets with a naive Bayes classifier," presented at the Presented at MedInfo, San Francisco, 2004.
[21] P. Kanerva, *Sparse Distributed Memory*. Cambridge, MA: MIT Press, 1988.
[22] D. Widdows and T. Cohen, "The semantic vectors package: New algorithms and public tools for distributional semantics," presented at the Fourth IEEE International Conference on Semantic Computing (IEEE ICSC2010), Carnegie Mellon University, Pittsburgh, Pennsylvania, 2010.
[23] D. Widdows and K. Ferraro, "Semantic vectors: a scalable open source package and online technology management application," in *Sixth International Conference on Language Resources and Evaluation*, 2008.
[24] (2012, 3-1-12). *Semantic Vectors Package*. Available: http://code.google.com/p/semanticvectors/
[25] D. Widdows, T. Cohen, and L. DeVine, "Real, complex, and binary semantic vectors," in *To appear in QI'12. Proceedings of the 6th International Symposium on Quantum Interactions*, Paris, France, 2012.
[26] E. Hatcher and Gospondnetic, O., *Lucene in action*. Greenwich, CT: Manning Publications Co., 2004.
[27] M. Wahle, D. Widdows, J. Herskovic, E. Bernstam, and T. Cohen, "Deterministic binary vectors for efficient automated indexing of MEDLINE/PubMed abstracts," presented at the Proc AMIA Symp, 2012.
[28] P. L. Schuyler, W. T. Hole, M. S. Tuttle, and D. D. Sheretz, "The UMLS metathesaurus: representing different views of biomedical concepts," *Bull Med Libr Assoc.,* vol. 81, pp. 217-222, 1993.
[29] H. Liu, "A Multi-aspect Comparison Study of Supervised Word Sense Disambiguation," *Journal of the American Medical Informatics Association,* vol. 11, pp. 320–331, 2004.
[30] M. Weeber, J. G. Mork, and A. R. Aronson, "Developing a test collection for biomedical word sense disambiguation," presented at the Proceedings / AMIA Annual Symposium., 2001.
[31] G. Salton, *The SMART retrieval system: experiments in automatic document processing*: Prentice-Hall, 1971.