Enhancing Medical Word Sense Inventories Using Word Sense Induction: A Preliminary Study

Qifei Dong[1][0000−0002−5751−3208] and Yue Wang[2][0000−0002−0278−2347]

[1] Department of Biomedical Informatics and Medical Education, University of Washington, Seattle WA 98195, USA
qfdong@uw.edu

[2] School of Information and Library Science, University of North Carolina at Chapel Hill, Chapel Hill NC 27599, USA
wangyue@email.unc.edu

Abstract. Correctly interpreting an ambiguous word in a given context is a critical step for medical natural language processing tasks. Medical word sense disambiguation assumes that all meanings (senses) of an ambiguous word are predetermined in a sense inventory. However, the sense inventory sometimes does not cover all senses or is outdated as new concepts arise in the practice of medicine. Obtaining all word senses is therefore the prerequisite work for word sense disambiguation. A classical method for word sense induction is string expansion, a rule-based method that searches the corpus for full forms of an abbreviation or acronym. Yet, it cannot be applied to ambiguous words that are not abbreviations. In this paper, we study methods that can semi-automatically discover word senses from a large-scale medical corpus, regardless of whether the word is an abbreviation. We conducted a comparative evaluation of four unsupervised data-driven methods, including context clustering, two types of word clustering, and sparse coding in word vector space. Overall, sparse coding outperforms the other methods. This demonstrates the feasibility of using sparse coding to discover more complete word senses. By comparing the senses discovered by sparse coding with those in senses inventory, we observed new word senses. For more than half of the ambiguous words in the MSH WSD data set (sense inventory maintained by National Library of Medicine), sparse coding detected more than one new word sense. This result shows an opportunity in enhancing medical word sense inventories with unsupervised data-driven methods.

Keywords: Medical word sense induction · Context clustering · Word clustering · Sparse coding

1 Introduction

Biomedical literature and clinical documents contain many ambiguous terms. For example, the word “mole” can represent a unit of amount of substance, a skin condition (nevus), and a type of mammal. The abbreviation “PCA” can

mean principal component analysis, patient-controlled analgesia, and prostate cancer, among many other meanings. Automatically assigning the correct meaning (a.k.a. sense) to an ambiguous word in a context is referred to as word sense disambiguation (WSD). WSD has received extensive research in the medical domain [5, 10, 18, 21], as it is an important step towards high-quality analysis of massive biomedical literature and clinical notes.

The first and foremost question for medical WSD is how to get all possible senses of an ambiguous word. Previous work either assumed that an existing knowledge base could provide these senses [9], or relied on human experts annotating many instances to obtain all possible senses [13]. However, the sense inventories generated from existing knowledge base sometimes do not cover all senses used in practice [9], and manual sense annotation requires specialized expertise and is time-consuming [13]. Limited amount of research also explored semi-automated approaches to discovering word senses from text corpus, i.e., word sense induction (WSI), which can discover diverse word senses with low annotation cost [22].

Two families of methods are usually used for WSI: data-driven and rule-based.

Three types of data-driven methods are commonly used. The first uses word contexts to obtain word senses. One way of using word contexts is context clustering. It starts by generating context vectors, each representing one instance of the target word's surrounding words. Then the context vectors are clustered into multiple groups, each representing a word sense. Such idea was first proposed by Schütze [19], who constructed the context vectors from second-order co-occurrence information. Researchers later employed the same idea with different context vector construction and clustering techniques [15, 17, 22, 23]. Purandare and Pedersen [15] proposed six context clustering systems, each using a different way of constructing and clustering context vectors. In the medical domain, Xu *et al.* [22, 23] used Expectation Maximization and Farthest First algorithms to cluster contexts. Savova *et al.* [17] applied the six context clustering systems proposed by Purandare and Pedersen [15] to the medical domain. Besides context clustering, some researchers created probabilistic models of the target word and its contexts. Brody and Lapata [4] proposed a Bayesian approach related to Latent Dirichlet Allocation and a generative model to find word senses. The third way of using contexts assumes the context's syntagmatic patterns are associated with the word senses. Then the main job is to find the context's syntagmatic patterns. Pustejovsky *et al.* [16] used Corpus Pattern Analysis to acquire the context patterns.

The second type of the data-driven methods is implemented on the word vector space. One common algorithm is to cluster the semantically similar words in the word vector space into the same group. To calculate semantic similarities, Pantel and Lin [14] employed pointwise mutual information. More recently, Arora *et al.* [2] studied the linear algebraic patterns in word embeddings and used sparse coding to obtain word senses.

Third, data-driven WSI can be conducted by graph-based methods. There, a graph of words is constructed and graph clustering algorithms are used to find word senses [1, 7].

One common algorithm for rule-based WSI is string expansion [20], which is mainly employed in abbreviation and acronym sense induction. Each of the abbreviation's senses can be represented by one of the abbreviation's full forms. To find the full forms, some rules (regular expressions) are set to expand the abbreviation. Then these regular expressions are used to match the abbreviation's full forms in corpora. Although the rule-based method has good precision and recall [20], this method is limited to finding word senses of abbreviations and acronyms.

In this study, we aimed to study WSI regardless of whether the word is an abbreviation/acronym. We therefore focused on the unsupervised data-driven methods for WSI. We conducted preliminary study on four unsupervised data-driven methods, including context clustering, two types of word clustering, and sparse coding. We adopted a novel evaluation method that measures the overlap of sense groups without human annotation. This allowed us to efficiently compare a wide range of the unsupervised data-driven methods. The result shows sparse coding outperforms the other methods, demonstrating the feasibility of using sparse coding to discover more complete word senses.

To evaluate the potential of the unsupervised data-driven methods in enhancing existing medical word sense inventories, we manually annotated the discovered senses. We took two well-established sense inventories: 1) the test collection, MSH WSD data set [9], derived from the Unified Medical Language System, Medical Subject Headings (MeSH), and MEDLINE abstracts; and 2) the clinical abbreviation inventory from the University of Minnesota [13]. The senses of each ambiguous word were discovered by running one of the unsupervised data-driven methods on a large-scale raw text corpus. We then compared the WSI-discovered senses against existing senses in the sense inventory. A systematic analysis on the WSI-discovered senses shows that in the MSH WSD data set, half of the ambiguous words are missing more than one major senses. This analysis demonstrates the unsupervised data-driven methods have great potential in enhancing existing sense inventories by finding new senses and associated contexts.

2 Method

In this section, we discuss the data sets, the unsupervised data-driven methods for WSI, the evaluation method, and how to interpret WSI-discovered senses. The overall workflow of our study is shown in Figure 1. Given an ambiguous word in a sense inventory and a large corpus, an unsupervised data-driven method for WSI discovered a set of senses. Then we compared the WSI-discovered word senses with the existing senses in the inventory, evaluated their overlap, and examined the new senses found by the unsupervised data-driven methods.

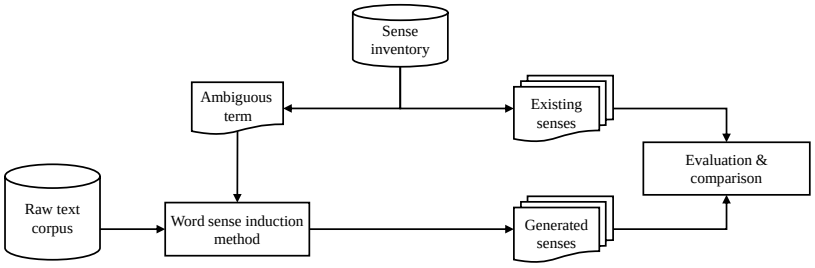

Fig. 1. Overall workflow.

To facilitate the evaluation and comparison, we represented a word sense as *a small set of semantically related words* and the set is called a *sense profile*. For example, one sense of the acronym "AB" is "AB influenza type," and its sense profile consists of a list of related words like "flu", "influenza", and "seasonal." This representation has several advantages over a natural language definition or an entry in a knowledge base. First, it is a flexible and faithful way of representing a sense, as a word shall be known "by the company it keeps [8]." Second, it does not assume the existence of a standardized superset of senses, which is why we set out finding new senses in the first place. Third, words in the sense profile can appear across corpora, making it possible to automatically and approximately estimate the overlap between senses learned from different corpora without resorting to expensive manual annotation. In our research, we set the number of words in each sense profile to 20.

2.1 Data Sets

We conducted our study on two genres of medical text: biomedical literature and clinical notes. Table 1 summarizes the basic information of the sense inventory and raw text corpus based on each of the genres. In this table, a context instance is defined by 20 words surrounding the ambiguous term (ten words each side).

Table 1. Statistics of the data sets.

		Biomedical literature	Clinical notes
Sense inventory	Name	MSH WSD data set	UMN clinical abbreviations
	# Ambiguous terms	184	75
	Avg. # context instances per ambiguous term	190	500
Raw text corpus	Name	MEDLINE abstracts	MIMIC-III clinical notes
	File size (Gigabytes)	13	4.6
	Avg. # context instances per ambiguous term	31,148	10,898

The sense inventories provide ambiguous terms, as well as their word senses and context instances. In each of these context instances, the corresponding term's word sense is known beforehand. Then for each word sense stored in the sense inventories, we could gather the context instances corresponding to this word sense. Using these gathered context instances, the sense profile of the word sense could be obtained. First, we computed the mutual information $I(s; w)$ between the sense s and any word w that appears in the gathered context instances. Then the 20 words with the highest $I(s; w)$ were selected to form the sense profile.

The sense inventories in the biomedical literature and the clinical note settings are the MSH WSD data set [9] and the University of Minnesota clinical abbreviation and acronym sense inventory (UMN) [13], respectively.

MSH WSD data set contains 203 ambiguous terms. In our experiments, we used 184 ambiguous terms. First, as our focus was single-word terms in this study, we excluded the multi-word terms. Including multi-word terms is our future direction. Second, the remaining terms with less than 100 context instances were excluded. The reason is to ensure robust estimation of mutual information $I(s; w)$ between a sense and a word. The average number of the context instances per ambiguous term is 190.

UMN contains 440 ambiguous terms. Each term is a single word. We excluded the terms with less than 100 context instances. 75 terms remained. The average number of the context instances per term is 500.

The raw text corpora are the data sets from which the unsupervised data-driven methods learn word senses of ambiguous terms. The raw text corpora in the biomedical literature and the clinical note settings are the MEDLINE abstracts and the admission notes in MIMIC-III, respectively. The average numbers of the context instances per term in the the MEDLINE abstracts and the MIMIC-III clinical notes are 31,148 and 10,898, respectively.

2.2 Unsupervised Data-Driven Methods for WSI

This section describes the four unsupervised data-driven methods for WSI. Given an ambiguous word and a large corpus, the goal of the methods is to discover a set of senses in this corpus, with each sense represented by a sense profile. Among the four methods, sparse coding and the two word clustering methods require dense word vectors. We trained 100-dimensional dense word vectors using the skip-gram algorithm [11] in Google's word2vec package for single words in a case-insensitive manner. In the biomedical literature setting, we trained word vectors using MEDLINE abstracts. In the clinical note setting, we trained word vectors using MIMIC-III clinical notes.

Each of the unsupervised data-driven methods contains hyper-parameters. As a preliminary study, we intuitively explored how to select the hyper-parameters' values. A comprehensive sensitivity analysis is our future direction.

Context Clustering The intuition of context clustering is that words with similar contexts are semantically similar to each other [12]. First, we extracted

context windows with ten words on both sides of an ambiguous word in the corpus. The last row of Table 1 shows that such context windows are abundant. After finding all context windows of the word, we performed *tf-idf* weighting to obtain sparse context vectors. Then we ran k-means to cluster these sparse vectors. Each resulting cluster formed one word sense. k is a hyper-parameter, determining the number of senses we expected the method to find for each ambiguous word. In this and the following methods that use k-means, k has the same meaning. We evaluated different settings of k in the experiments (Section 3). To get the sense profile of each sense, we calculated the centroid of the corresponding cluster and selected the 20 words with the highest weights in the centroid vector.

Word Clustering I (Nearest Words in Context Windows) According to the distributional hypothesis [8], words tend to have related senses if they occur in similar context. First, we extracted context windows in the same way as the previous method. Then we took out all words appearing in these context windows, except the target word itself and the stopwords like "the" and "of."[1] Each extracted word could be represented by a dense word vector. We ran k-means on these dense word vectors and obtained k centroids. The 20 words whose dense word vectors were closest to the centroid were used as the words in the sense profile. The distance between two vectors in the word vector space was measured by cosine distance.

Word Clustering II (Nearest Words in Vector Space) This algorithm obtains word senses directly in word vector space. As nearby words in word vector space are semantically related to each other, the senses of a target word could be contained in its neighbors. For the same reason, these neighboring words cannot distribute evenly in space. Instead, they should cluster into groups to form senses.

To get the k senses of a target word, we ran k-means on the target word's N nearest neighbors in word vector space. Then for each sense, we selected 20 words whose dense word vectors are closest to the corresponding cluster centroid as the words in the sense profile. Note an appropriate value of N is important. If N is too small, we could miss certain senses. If N is too large, we could obtain irrelevant senses. We empirically set $N = 500$ as a reasonable size.

Sparse Coding We adopted the sparse coding method from Arora *et al.*'s work [2]. Sparse coding works directly in word vector space, and is based on the assumption that each word is a linear combination of some word cluster centroids [2]. Each of these clusters contains several words and forms a sense. For example, the word *BAT* could mean *Chiroptera*, a kind of mammal, as well as *Brown Fat*. Let d denote the number of dimensions of the word vectors, $\boldsymbol{v}_{BAT} \in \mathbb{R}^d$ denote the word vector of *BAT*, and $\boldsymbol{c}_{BAT1}, \boldsymbol{c}_{BAT2} \in \mathbb{R}^d$ denote

[1] The full list of stopwords is available at `https://www.ranks.nl/stopwords` .

the cluster centroids representing the two senses of *BAT*, respectively. Ideally, $\boldsymbol{v}_{BAT} = r_{BAT1}\boldsymbol{c}_{BAT1} + r_{BAT2}\boldsymbol{c}_{BAT2}$, where r_{BAT1} and r_{BAT2} are coefficients.

Let m denote the total number of word clusters in the whole word vector space. m needs to be pre-defined before conducting sparse coding. We set $m =$ 2,000 as suggested in the previous work [2]. Let $\left\{\boldsymbol{c}_i \in \mathbb{R}^d\right\}_{i=1}^{m}$ denote the set of the word cluster centroids. Let V denote the vocabulary, i.e., the set of all words in a corpus. When representing the word $w_j \in V$, we use $r_{j,i}$ to denote the coefficients multiplying $\boldsymbol{c}_i$. Let $\boldsymbol{\epsilon}_j \in \mathbb{R}^d$ denote a noise vector. Then for any word $w_j \in V$, its word vector can be represented by $\sum_{i=1}^{m} r_{j,i}\boldsymbol{c}_i + \boldsymbol{\epsilon}_j$. As a word can only have a small number of senses, most of the coefficients $r_{j,i}$ should be zero, hence the name *sparse* coding. Each of the cluster centroids multiplied by a non-zero coefficient represents one of the target word's senses. Our goal is to 1) find all of the word clusters in the word vector space, and 2) for each target word, obtain the cluster centroids with non-zero coefficients. The number of the cluster centroids with non-zero coefficients is a pre-defined hyper-parameter and denoted by k. k determines the number of senses we expected sparse coding to find for each ambiguous word. This k intrinsically has the same meaning as k in k-means mentioned above does. When implementing the sparse coding method, we varied k for different medical text corpora. For each resulting sense, we selected 20 words whose dense word vectors are closest to the corresponding cluster centroid as the words in the sense profile.

2.3 Evaluation Method

As the senses discovered by the unsupervised data-driven methods are represented by the sense profiles rather than human readable labels, we need a method for telling whether a discovered sense matches any actual sense. We adopted the novel evaluation method called "police lineup" proposed by Arora *et al.* [2]. This evaluation method estimates the degree of overlap between two sets of senses without resorting to labeling context instances in corpora. It allows efficient evaluation of different unsupervised data-driven methods against a sense inventory. Intuitively, it tests how well the WSI-discovered senses match the actual senses and distinguish from the irrelevant ones. It is called "police lineup" because it is analogous to the investigation process where a witness has to identify the suspect from several innocent people.

We illustrate the police lineup evaluation in Figure 2. This figure depicts a word vector space, where semantically related words are close to each other. The pentagon is the target word, associated with the vector $\boldsymbol{v}_t$. S_1-S_6 represent senses existing in the sense inventory. S_1, S_2, and S_3 are the target word's actual senses in the sense inventory. S_4, S_5, and S_6 do not belong to the target word, and hence are termed *distracting senses*. The solid circles inside each of S_1-S_6 represent the words in the corresponding sense profile. Each of the 5-point stars $\boldsymbol{c}_1$, $\boldsymbol{c}_2$, and $\boldsymbol{c}_3$ represents a WSI-discovered sense, obtained by calculating the mean of the word vectors in the corresponding sense profile. The police lineup evaluation asks each of the WSI-discovered senses to return its closest existing sense(s). In Figure 2, $\boldsymbol{c}_1$ picks S_1, $\boldsymbol{c}_2$ picks S_5, and $\boldsymbol{c}_3$ picks S_2. The picked senses are called

candidate senses. Then precision and recall can be calculated. Precision is the number of the actual senses picked out divided by the number of the candidate senses. Recall is the number of the actual senses picked out divided by the total number of the actual senses of the target word.

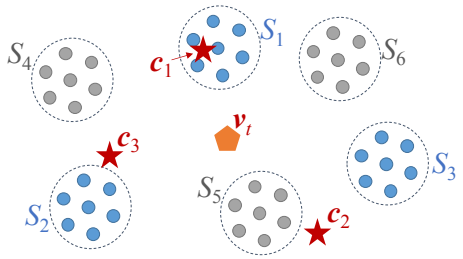

Fig. 2. The police lineup evaluation method.

In practice, we randomly selected some of the existing senses not attributed to the target word as the distracting senses. The number of the selected distracting senses should be set properly. If this number is too large, the senses discovered by either mediocre or worse WSI method could hardly pick the actual senses. As a result, we could hardly compare the performance of these methods using police lineup evaluation. If this number is too small, the senses discovered by either mediocre or better WSI method could accurately pick the actual senses. In our study, given a target word, the number of its distracting senses was set to 20 minus the number of the actual senses of the target word.

Overlap score between a WSI-discovered sense and an existing sense is used to decide which existing sense(s) the WSI-discovered sense should pick. Let C_t denote the set of WSI-discovered senses of the target word. Each element $\boldsymbol{c} \in C_t$ represents a WSI-discovered sense, obtained by getting the mean of the word vectors in the corresponding sense profile. Let n denote the number of words in the sense profile. Let d denote the number of dimensions the word vectors have. Let T_t denote the set containing the actual and distracting senses. Each element $\mathbf{S} \in T_t$ is a matrix in $\mathbb{R}^{n \times d}$ representing one sense in T_t. Each row in $\mathbf{S}$ is a vector representing a word in the sense profile of this sense. The row order does not matter. Let $\|\cdot\|_2$ denote the L_2 norm of a vector. Let $|V|$ denote the number of words in the vocabulary V. We calculate the overlap score between a WSI-discovered sense $\boldsymbol{c}$ and a sense $\mathbf{S}$ for the target word using the following formula:

$$score_t(\boldsymbol{c}, \mathbf{S}) = \left(\|\mathbf{S}\boldsymbol{c}\|_2 - \frac{1}{|C_t|} \sum_{\boldsymbol{c}' \in C_t} \|\mathbf{S}\boldsymbol{c}'\|_2 \right) + \left(\|\mathbf{S}\boldsymbol{v}_t\|_2 - \frac{1}{|V|} \sum_{j=1}^{|V|} \|\mathbf{S}\boldsymbol{v}_j\|_2 \right). \tag{1}$$

The two parts measure how close $\boldsymbol{c}$ is to $\mathbf{S}$ and how close $\mathbf{S}$ is to $\boldsymbol{v}_t$, respectively. The second part is to prevent the cases that both the WSI-discovered sense and

the sense in T_t are too far away from the target word, which can happen if the sense in T_t is a distracting one. The two subtracted terms are average similarities. Algorithm 1 describes the process to pick candidate senses from the actual and distracting senses.

Algorithm 1 Police lineup evaluation

```
Initialize: an empty set used to contain candidate senses, candidates
for each cluster centroid c ∈ C_t do
    for each sense S in the set T_t do
        Calculate the overlap score between c and S using Equation (1)
    Let U := {top 2 highest-scoring S}
    candidates ← candidates ∪ U
return Top p senses with the highest scores in candidates
```

By varying p, we got different precision and recall for detecting the actual senses out of the candidate senses. We varied p from 1 to the maximum number of elements that the set *candidates* could have, $2 \times |C_t|$. Then we drew the precision-recall curve. A larger area under the curve means the WSI-discovered senses cover more actual senses. As these actual senses are typical senses stored in sense inventories, finding more actual senses indicates the unsupervised data-driven method is more reliable.

2.4 Interpreting WSI-Discovered Senses

Ideally, we could interpret a WSI-discovered sense using the words in the corresponding sense profile. Yet, it can be a tough task as these words are professional and difficult for interpretation. We used two approaches to better interpret the WSI-discovered senses. First, given a word in the sense profile, we printed out its semantically related words that are more commonly used and more familiar to laymen. These common words were also added to the sense profile. We term the original 20 words in the sense profile *precise sense-profile words*, and the common ones *common sense-profile words*. Second, given one WSI-discovered sense of a target word, we extracted the context instances where the target word bears this sense from MEDLINE abstracts or MIMIC-III clinical notes.

Identifying Common Sense-Profile Words In word vector space, the frequently used words close to the precise sense-profile words were found. These frequently used words are the common sense-profile words. Algorithm 2 describes the detail. The number of common sense-profile words returned is 20.

Extracting Representative Context Instances Another way to interpret a WSI-discovered sense is to look at the context instances where the ambiguous word most likely to bear that sense. Hence, we need a method to automatically

Algorithm 2 Finding common sense-profile words

1: **Input**:
   1) Precise sense-profile words of a sense. Let S denote the set containing the precise sense-profile words and $s \in S$ denote one of them. Let n denote the number of the precise sense-profile words in each sense profile
   2) A corpus
   3) Word vectors
2: Given the corpus, rank all words by their frequency from high to low. Take out the words from Rank 1 – 8000
3: Group these words by their ranking. Specifically, words from Rank 1 – 1000, 1001 – 2000, 2001 – 4000 and 4001 – 8000 are grouped to 4 sets, respectively denoted by G_1, G_2, G_3, and G_4
4: Initialization: Let Out denote the output set, which is initialized to an empty set
5: **for** $i = 1, 2, 3, 4$ **do**
6: **for** each word $w \in G_i$ **do**
7: $score(w, S) = \frac{1}{n} \sum_{s \in S} \cos(\boldsymbol{v}_w, \boldsymbol{v}_s)$
8: Let $U :=$ {top 5 highest-scoring w}
9: $Out \leftarrow Out \cup U$.
10: **return** Out

infer what the ambiguous word means in a certain context instance. To do this task, we used the precise sense-profile words of the WSI-discovered senses.

A context instance x consists of a set of words $\{w_i\}_{i=1}^m$ (excluding the target word itself), where m is the number of words in this set. $\boldsymbol{v}_i$ denotes the word vector of w_i in this set. Let Y be the set of the WSI-discovered senses. A sense $y \in Y$ is represented by a set of precise sense-profile words $\{w_j\}_{j=1}^n$, where n is the number of the words. $\boldsymbol{v}_j$ denotes the word vector of the precise sense-profile word w_j. The relatedness score between x and y is evaluated by

$$r(x, y) = \sum_{j=1}^{n} \max_{1 \le i \le m} \cos(\boldsymbol{v}_i, \boldsymbol{v}_j).$$

This formula means given any word representing the sense y, if the context instance x contains at least one word semantically similar to the given word, the context instance x is related to the sense y. The scores can be converted into a probability distribution of senses: $p(y|x) = \exp[r(x, y)] / \sum_{y' \in Y} \exp[r(x, y')]$. To interpret sense y, we can look at those context instances with the highest $p(y|x)$ and examine the sense of the ambiguous word in those contexts.

Note even with the above two approaches (showing the common sense-profile words and representative context instances), we could still occasionally fail to interpret WSI-discovered senses due to low-quality clustering results. We call such WSI-discovered senses "unclear senses."

3 Results

Our experiments ran on a MacBook Pro laptop with one four-core Intel Core i7-7700HQ 2.8GHz central processing unit, 16GB memory, one 256GB Mac-

intosh HD disk, and running the macOS High Sierra 10.13 operating system. The context clustering method was written in Python 3.6. The other unsupervised data-driven methods and the evaluation method were written in MATLAB R2017b. Sparse coding was solved by a MATLAB toolbox called SMALLbox [6].

3.1 Precision-Recall Curves

We conducted each unsupervised data-driven method twice on each genre of the medical text (biomedical literature or clinical notes). Each time, we set k to a different value. Recall k is the number of word senses we expected a method to find for one ambiguous term. In the biomedical literature setting, we set k to 4 and 5. In the clinical note setting, k was set to 5 and 8. k was set larger in the clinical note setting because the average number of actual senses of a clinical abbreviation is larger. To compare the methods, we used the police lineup evaluation to generate the precision-recall curves, shown in Figure 3. Table 2 lists the area under each precision-recall curve.

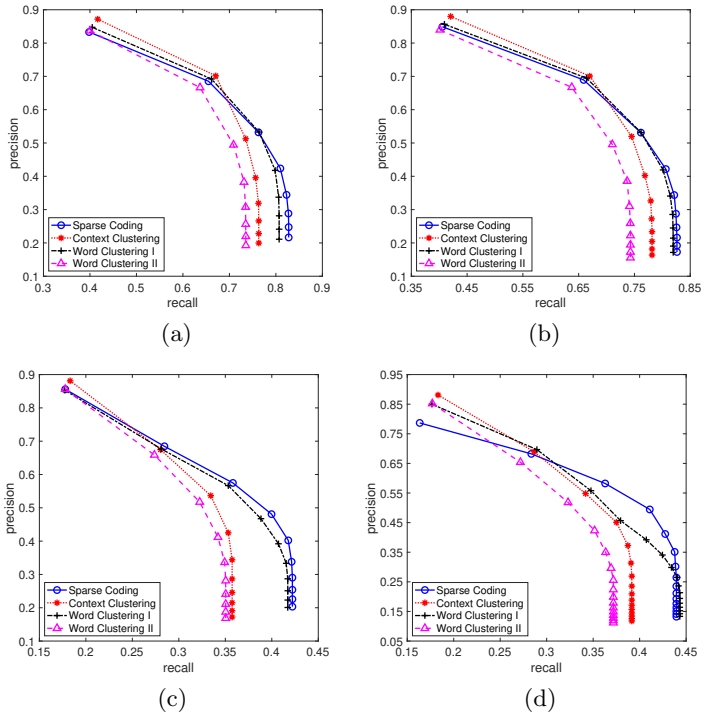

Fig. 3. Given different genres of medical text and k, the precision-recall curves for the unsupervised data-driven methods. (a) $k = 4$ in the biomedical literature setting. (b) $k = 5$ in the biomedical literature setting. (c) $k = 5$ in the clinical note setting. (d) $k = 8$ in the clinical note setting.

Table 2. Area under the precision-recall curve.

	Biomedical literature		Clinical notes	
	$k = 4$	$k = 5$	$k = 5$	$k = 8$
Sparse coding	**0.290**	**0.286**	**0.160**	**0.176**
Context clustering	0.251	0.258	0.119	0.138
Word clustering I	0.280	0.276	0.147	0.174
Word clustering II	0.231	0.234	0.113	0.122

Figure 3 shows sparse coding outperforms the other methods across all tests, especially at the high recall end. Tables 2 shows sparse coding has the largest area under the curve in each test.

3.2 Case Studies of the Word Senses Discovered by Sparse Coding

In this section, we show case studies of the word senses discovered by sparse coding on the biomedical literature setting. We provide the sense profiles and context instances of the word *EPI*. We also list some WSI-discovered senses not stored in the existing inventories.

Sense Profile Table 3 shows a WSI-discovered sense of the word *EPI* with the sense profile. By looking at both precise and common sense-profile words, we can infer that this sense is related to hormone. By further consulting representative context instances, we know the sense is *Epinephrine*, a kind of hormone.

Table 3. A WSI-discovered sense of the word *EPI* and the sense profile (only the ten most representative precise and common sense-profile words are shown).

Precise sense-profile words	Common sense-profile words	Annotated sense
crh trh adrenocorticotropin corticotropin beta-end thyrotrophin-releasing crf-41 beta-endorphin acth corticoliberin	hormone insulin secretion gh pituitary dopamine acth prl prolactin trh	Epinephrine

Representative Context Instances Table 4 shows context instances for another sense of *EPI*, with estimated $p(y|x)$ and PubMed Identifiers (PMID). It is clear that *EPI* means *Echo-Planar Imaging* here. This sense is not included in the MSH WSD data set.

We further estimated the frequency of the newly discovered sense to see whether it is a major sense. Take *EPI = echo planar imaging* as an example, there were 614 context instances in the MEDLINE abstracts with $p(y|x) \geq 0.960$. We randomly selected 50 out of the 614 context instances and found that in 48 (96%) of them, *EPI* indeed means *Echo-Planar Imaging*. Therefore, this is a major sense of *EPI* used in hundreds of the MEDLINE abstracts. We did the

Table 4. Example context instances associated with the sense *Echo-Planar Imaging*.

PMID	$p(y\|x)$	Context instance
12417991	1.000	*. . . within the brain, causing geometric distortions in echo planar imaging (EPI). Even if subtle, change in shim can lead to artifactual . . .*
9332249	0.984	*. . . each of which consists of a number of gradient echoes (EPI factor, EF). The aim of our studywas to evaluate . . .*
15670684	0.960	*. . . brain activation can be monitored during the ongoing scan. However, EPI suffers from geometric distortions due to inhomogeneities of the magnetic . . .*

same trial on the newly discovered senses of other words. Most of the newly discovered senses are major senses.

Newly Discovered Senses Sparse coding identified a large number of newly discovered senses. Table 5 shows some examples. In this table, "existing senses" are provided by the MSH WSD data set. "WSI-discovered senses" can be further divided into two groups: 1) senses included in the existing senses ("overlapping existing senses"), and 2) senses not included in the existing senses ("newly discovered"). We also documented the number of unclear senses in this table. Across all 184 ambiguous terms in the MSH WSD data set, the average sense overlap is 61.4% per word. Sparse coding found the new senses for 100 ambiguous words, and a total number of 162 new senses were found. This means more than half of the words in the MSH WSD data set miss at least one major sense (1.62 senses to be exact). Our analysis results, including the sense profiles and the senses inferred using the context instances, are available at `http://bit.ly/2Id6837`.

Table 5. Comparison between the senses stored in the MSH WSD data set and the WSI-discovered senses.

Ambiguous term	Existing senses	WSI-discovered senses		Number of unclear senses
		Overlapping existing senses	Newly discovered	
Epi	Epirubicin; Epinephrine	Epirubicin; Epinephrine	Echo-planar imaging; Extended program of immunization	1
Moles	Talpidae; Nevus	Talpidae; Nevus	Mole, unit of measurement; Hydatidiform mole	0

4 Discussion

Overall, sparse coding outperforms the other three methods, especially at the high recall end. This means in most cases, using sparse coding can discover more

complete word senses from large-scare medical text. Sparse coding computes a global set of senses, while the word clustering methods compute senses for each word locally. Sparse coding outperforming word clustering indicates medical words may have very diverse and uncorrelated senses that do not belong to a local region of the word semantic space. Both sparse coding and word clustering I perform better than context clustering. This indicates clusters in dense word vector space could better represent a sense than clusters consisting of sparse context vectors.

Our analysis shows the MSH WSD data set can be enhanced using the unsupervised data-driven methods. The MSH WSD data set missing major senses could result from some problems occurred in the construction steps of this data set [9]. The MSH WSD data set was constructed by three steps. First, Unified Medical Language System (UMLS) [3] was screened to get ambiguous terms. Each term was linked to some MeSH terms, which served as senses. Second, each ambiguous term and its related MeSH terms were used to extract MEDLINE citations, which served as context instances. Third, the researchers eliminated trivial and repeated senses using three filters. The first filter removed the trivial senses whose corresponding context instances were very few. By conducting Support Vector Machine on the extracted MEDLINE citations of each ambiguous term, the second filter checked whether some of the term's senses were semantically similar and removed the repeated ones, if any. The third filter removed single-letter terms. In our research, the unsupervised data-driven methods discovered two types of senses not in the MSH WSD data set. One type, like *Extended Program of Immunization*, is included in neither MeSH vocabulary nor UMLS. These senses could not be obtained in the first step of constructing the MSH WSD data set. The other type is included in MeSH vocabulary or UMLS, which means such senses were removed by the filters. Yet, some newly discovered senses like *Echo-Planar Imaging* and *Mole, Unit of Measurement* are frequently used and distinct from other senses, and hence should not have been removed. Two reasons could explain why such senses were filtered out: 1) the senses were not commonly used when the MSH WSD data set was constructed, and hence removed by the first filter; and 2) in the second filter, Support Vector Machine did not give sufficiently accurate classification results.

There are several directions for future work. First, we only systematically annotated the WSI-discovered senses of the ambiguous terms in the MSH WSD data set, because it is relatively easy to understand their contexts – biomedical literature. In the future, we aim to collaborate with domain experts to annotate the WSI-discovered senses for clinical abbreviations and compare the senses discovered by WSI to those in the clinical sense inventory. Second, unsupervised learning algorithms, especially flat clustering algorithms like k-means, could sometimes generate low-quality clusters. To improve the clusters' quality, a comprehensive sensitivity analysis should be conducted to obtain proper ranges of the hyper-parameters in these algorithms. Also, we will try more powerful clustering algorithms like Tight Clustering for Rare Senses [23]. Third, the cur-

rent work did not include multi-word ambiguous terms. Exploring these terms is another future direction.

5 Conclusion

In this paper, we did a preliminary study on the four unsupervised data-driven methods for WSI, including context clustering, two types of word clustering, and sparse coding. We applied these unsupervised data-driven methods on two genres of medical text, biomedical literature and clinical notes. Among the four methods, sparse coding outperforms the other three methods, showing the feasibility of using sparse coding to discover more complete word senses from large-scare medical text. We analyzed the senses discovered by the unsupervised data-driven methods against those in the existing sense inventories. Our analysis showed that the sparse coding method detected more than one major sense for more than half of the ambiguous words in the MSH WSD data set. This result demonstrates that it is very promising to employ the unsupervised data-driven methods to improve sense coverage in the existing sense inventories.

References

1. Agirre, E., Martínez, D., de Lacalle, O.L., Soroa, A.: Two graph-based algorithms for state-of-the-art wsd. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. pp. 585–593. Association for Computational Linguistics, Sydney, Australia (2006)
2. Arora, S., Li, Y., Liang, Y., Ma, T., Risteski, A.: Linear algebraic structure of word senses, with applications to polysemy. Transactions of the Association for Computational Linguistics **6**, 483–495 (2018)
3. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic Acids Research **32**(suppl_1), D267–D270 (2004)
4. Brody, S., Lapata, M.: Bayesian word sense induction. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. pp. 103–111. Association for Computational Linguistics, Athens, Greece (2009)
5. Chen, Y., Cao, H., Mei, Q., Zheng, K., Xu, H.: Applying active learning to supervised word sense disambiguation in MEDLINE. Journal of the American Medical Informatics Association **20**(5), 1001–1006 (2013)
6. Damnjanovic, I., Davies, M.E., Plumbley, M.D.: SMALLbox-an evaluation framework for sparse representations and dictionary learning algorithms. In: Proceedings of the 9th International Conference on Latent Variable Analysis and Signal Separation. pp. 418–425. Springer, St. Malo, France (2010)
7. Di Marco, A., Navigli, R.: Clustering and diversifying web search results with graph-based word sense induction. Computational Linguistics **39**(3), 709–754 (2013)
8. Firth, J.R.: A synopsis of linguistic theory, 1930-1955. Studies in Linguistic Analysis (1957)
9. Jimeno-Yepes, A.J., McInnes, B.T., Aronson, A.R.: Exploiting MeSH indexing in MEDLINE to generate a data set for word sense disambiguation. BMC Bioinformatics **12**(1), 223 (2011)

10. Liu, H., Teller, V., Friedman, C.: A multi-aspect comparison study of supervised word sense disambiguation. Journal of the American Medical Informatics Association **11**(4), 320–331 (2004)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. pp. 3111–3119. Curran Associates Inc., Lake Tahoe, Nevada, United States (2013)
12. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. Language and Cognitive Processes **6**(1), 1–28 (1991)
13. Moon, S., Pakhomov, S., Liu, N., Ryan, J.O., Melton, G.B.: A sense inventory for clinical abbreviations and acronyms created using clinical notes and medical dictionary resources. Journal of the American Medical Informatics Association **21**(2), 299–307 (2013)
14. Pantel, P., Lin, D.: Discovering word senses from text. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 613–619. Association for Computing Machinery, Edmonton, Canada (2002)
15. Purandare, A., Pedersen, T.: Word sense discrimination by clustering contexts in vector and similarity spaces. In: Proceedings of the 8th Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004. Association for Computational Linguistics, Boston, MA, USA (2004)
16. Pustejovsky, J., Hanks, P., Rumshisky, A.: Automated induction of sense in context. In: Proceedings of the 20th international conference on Computational Linguistics. pp. 924–930. COLING, Geneva, Switzerland (2004)
17. Savova, G., Pedersen, T., Purandare, A., Kulkarni, A.: Resolving ambiguities in biomedical text with unsupervised clustering approaches. University of Minnesota Supercomputing Institute Research Report (2005)
18. Schuemie, M.J., Kors, J.A., Mons, B.: Word sense disambiguation in the biomedical domain: an overview. Journal of Computational Biology **12**(5), 554–565 (2005)
19. Schütze, H.: Automatic word sense discrimination. Computational Linguistics **24**(1), 97–123 (1998)
20. Siklósi, B., Novák, A., Prószéky, G.: Resolving abbreviations in clinical texts without pre-existing structured resources. In: 4th Workshop on Building and Evaluating Resources for Health and Biomedical Text Processing (2014)
21. Xu, H., Markatou, M., Dimova, R., Liu, H., Friedman, C.: Machine learning and word sense disambiguation in the biomedical domain: design and evaluation issues. BMC Bioinformatics **7**(1), 334 (2006)
22. Xu, H., Stetson, P.D., Friedman, C.: Methods for building sense inventories of abbreviations in clinical notes. Journal of the American Medical Informatics Association **16**(1), 103–108 (2009)
23. Xu, H., Wu, Y., Elhadad, N., Stetson, P.D., Friedman, C.: A new clustering method for detecting rare senses of abbreviations in clinical notes. Journal of Biomedical Informatics **45**(6), 1075–1083 (2012)