Word Sense Induction Using Lexical Chain based Hypergraph Model

Tao Qian[1,3], **Donghong Ji**[*1], **Mingyao Zhang**[2], **Chong Teng**[1], **and Congling Xia**[1]
(1) Computer School, Wuhan University, Wuhan, China
（2）College of Foreign Languages and Literature, Wuhan University, Wuhan, China
(3) College of Computer Science and Technology, Hubei University of Science and Technology,XianNing, China
{taoqian, dhji, myzhang, tengchong, clxia}@whu.edu.cn

Abstract

Word Sense Induction is a task of automatically finding word senses from large scale texts. It is generally considered as an unsupervised clustering problem. This paper introduces a hypergraph model in which nodes represent instances of contexts where a target word occurs and hyperedges represent higher-order semantic relatedness among instances. A lexical chain based method is used for discovering the hyperedges, and hypergraph clustering methods are used for finding word senses among the context instances. Experiments show that this model outperforms other methods in supervised evaluation and achieves comparable performance with other methods in unsupervised evaluation.

1 Introduction

Word sense induction (WSI) aims to automatically find senses of a given target word (Yarowsky, 1995) from large scale texts. Compared with existing manual word sense resources, WSI techniques use clustering algorithms to determine the possible senses for a word.

Word sense induction is generally considered as an unsupervised clustering problem. The input for the clustering algorithm is context instances of a target word, represented by word bags or co-occurrence vectors, and the output is a grouping of these instances into classes, each corresponding to an induced sense.

Traditional methods in WSI tend to adopt the vector space model, in which the context of each instance of a target word is represented as a vector of features based on frequency statistics and probability distributions, e.g., first-order or second-order vector (Schütze, 1998; Purandare and Pedersen, 2004; Cruys et al., 2011). These vectors are clustered and the resulting clusters represent the induced senses. Another family of employed approach is graph-based methods (Widdows and Dorow, 2002; Véronis, 2004; Agirre et al., 2006; Klapaftis and Manandhar, 2007; Di Marco and Navigli, 2011; Hope and Keller, 2013), which have been recently explored successfully to some extent. Graph-based methods are considering the notion of a co-occurrence graph, assuming a binary relatedness between co-occurring words.

One of the key challenges in WSI is learning the higher-order semantic relatedness among multiple context instances. Previous approaches (Klapaftis and Manandhar. 2007; Bordag, 2006) for WSI are used to construct higher-order relatedness by counting co-occurrence frequency or collocation of mutiwords, regardless of global semantic similarity.

Lexical chain (Morris and Hirst, 1991) is defined as a sequence of semantically related words in text and provides important clues about the text structure and topic. It can be viewed as a global counterpart of the measures of semantic similarity (Navigli, 2009). For example, Figure 1 gives three context instances containing Apple.

* Corresponding author E-mail: dhji@whu.edu.cn.

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1601–1611, Dublin, Ireland, August 23-29 2014.

i) **Apple** designs and creates *iPod* and *iTunes*.
ii) The **Apple** support homepage is your starting point for help with **Apple** *hardware and software products*.
iii) Get detailed market's price information for the *products* of **Apple** *Inc*.

Figure 1. Context instances of Apple

Obviously, four Apples in Figure 1 all refer to the Apple Company. We can directly group three instances by the lexical chain: *iPod-iTunes-hardware and software product-Inc*. This lexical chain represents a higher-order semantic relatedness among the three instances.

In this paper, we propose a hypergraph model from a *global* perspective, in which nodes represent instances of contexts where a target word occurs and hyperedges denote higher-order semantic relatedness among instances. A lexical chain based method is used for identifying the hyperedges. This method for lexical chain extraction is a knowledge-free method based on LDA topic model (Remus and Biemann, 2013).

The remainder of this paper is structured as follows. Section 2 presents an overview of the related work. Section 3 describes our model in details. Section 4 provides a quantitative evaluation and comparison with other algorithms in the SemEval-2013 word sense induction task. Finally, section 5 draws conclusions and lays out some future research directions.

2 Related Work

2.1 Word sense induction

A number of diverse approaches to WSI have been proposed so far. Context features are often represented in a variety of forms such as co-occurrence of words within phrases (Pantel and Lin, 2002; Dorow and Widdows, 2003), parts of speeches (Purandare and Pedersen, 2004), and grammatical relations (Pantel and Lin, 2002; Dorow and Widdows, 2003). The size of the context window also varies, such as two words before and after the target word, the sentence or even larger paragraph within which contains the target word.

Most of the work in WSI is the vector space model, such as context-based vector algorithm (Schütze, 1998; Ide et al., 2001; Van de Cruys et al., 2011), substitute-based vector algorithm (Yatbaz et al., 2012; Baskaya et al., 2013). In this model, the context of each instance of a target word is represented as a vector of features based on frequency statistics or probability distributions (e.g., first-order or second-order vector). These vectors are clustered by various algorithms and the resulting clusters represent the induced senses.

Another family of employed approach is graph-based methods, which have been successfully applied in the sense induction task with some better results achieved. In this framework words are represented as nodes in the graph and vertices are drawn between the target word and its co-occurrences. The co-occurrences between words can be obtained on the basis of grammatical (Widdows and Dorow, 2002) or collocational relations (Véronis, 2004). Senses are induced by identifying highly dense sub-graphs (hubs) in the co-occurrence graph.

Klapaftis (2007) uses hypergraph model for WSI, in which co-occurrences of two or more words are represented by using weighted hyperedges. This model fully exploits the existence of collocations or terms consisting of more than two words. In fact, the method converts the sense induction problem to the clustering of the contextual words, and the result relies on local word co-occurrence frequency. Our hypergraph model is constructed from a global perspective, where the whole context instance is regarded as a node.

WSI evaluation also is an important issue in WSI tasks. Previous WSI evaluations in SemEval (Agirre and Soroa, 2007; Manandhar et al., 2010) have approached sense induction in terms of finding the single most salient sense of a target word given its context. However, as shown in Erk and McCarthy (2009), multiple senses of the target word may be perceived by readers from different angles and a graded notion of sense labeling may be considered as the most appropriate. The SemEval-2013 WSI evaluation is designed to explore the possibility of finding all perceived senses of a target word in a single context instance. Our model is evaluated and verified on the SemEval-2013 WSI task.

Algorithm 1. lexical chains extraction algorithm

Input: training set D of target word, hyper-parameters of LDA model; semantic threshold γ.
Output: lexical chain set S

```
θ,φ,Z ← LDA (D)
for each topic z
  lc =""              // lc denotes a lexical chain
  for each doc d
    for each word w in doc d
      if ( z_w = z and p(w,d|z) > γ )
        lc.add (w)
S.add (lc)
return S
```

2.2 Lexical chain extraction

The Lexical chain method is an important technique in natural language processing. A lexical chain is a sequence of semantic related words in text and provides important clues about the text structure and topic. It has formed a theoretically well-founded building block in a lot of applications, such as word sense disambiguation (Manabu and Takeo, 1994), malapropism detection and correction (Hirst and St-Onge, 1998), summarization (Barzilay et al., 1997), topic tracking (Carthy, 2004), text segmentation (Stokes et al., 2004), and others.

There are mainly two approaches for lexical chain extraction. One focuses on the use of knowledge resources like WordNet (Hirst and St-Onge, 1998) or thesauri (Morris and Hirst, 1991) as background information in order to quantify semantic relations between words. A major disadvantage of this strategy is that it relies on the resource, which has a direct impact on the quality of lexical chains. Another approach is based on statistical methods (Remus and Biemann, 2013). In this paper, we follow Remus and Biemann (2013) to automatically extract lexical chain by using LDA topic model.

3 Hypergraph model

In general, lexical chain based hypergraph model contains the following steps:

i) Automatically extracting lexical chains based on topic model;
ii) Constructing hypergraph with lexical chains;
iii) Inducing word sense by hypergraph clustering.

3.1 Lexical chain extraction

The extraction technique of lexical chains is based on LDA topic model. LDA topic model (Blei et al., 2003) is a probabilistic model of text generation designed for revealing some hidden structure in large data collections. The key idea is that each document can be represented as a probability distribution over a fixed set of topics where each topic can be represented as a probability distribution over words. We use LDA topic model for estimating the semantic closeness of lexical terms, and explore a way of utilizing LDA's topic information in constructing lexical chains automatically. In our model, document is replaced with context instance of a target word.

We adopt the idea of interpreting lexical chains as topics and placing all word tokens that share the same topic into the same chain. Lexical chains are usually extracted from the same paragraph or text, whose topic distributions are identical. However, in our experiment the context instances of a target word for WSI are derived from different articles, whose topic distributions are varied. Therefore both lexical and contextual topics are modeled. After training the LDA model, we use the information of the per-document topic distribution $\theta_d = p(z|d)$, the per-topic word distribution $\phi_w = p(w|z)$ and the sampling topic of a word z_w.

The key work lies in how to assign a word to a topic in training LDA model. Since single samples of topics per word may exhibit a large variance (Riedl and Biemann, 2012), we sample several times and use the mode (most frequently assigned) topic ID per word as the topic assignment.

d_1. WOMEN are half the population but only 15% of board members at big American firms, and 10% in Europe.

d_2. He referred the dispute to the board of directors.

d_3. The chairman of your board wears a blue shirt.

d_4. Your board should be involved in helping you sell and buy candidates.

d_5. The boy pounded the nails into the board.

d_6. A board is a flat, thin, rectangular piece of wood used for a particular purpose.

d_7. The board should have ropes attached to each end so you can loop it over your neck.

(a)

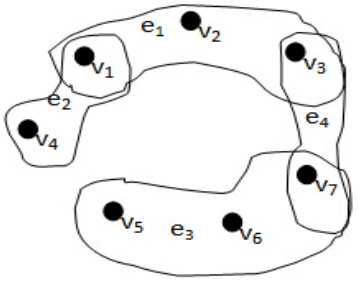
Chain 1: (members#d_1,directors#d_2,chairman#d_3)

Chain 2: (firms#d_1, candidates#d_4)

Chain 3: (nails#d_5, ropes#d_6, wood#d_7)

Chain 4: (skirt#d_3, neck#d_7)

(b)



(c)

Figure 2. An example of hypergraph creation. (a). seven context instances of **board**; (b). four lexical chains extracted; (c). the created hypergraph. The d_i in (a) corresponds to the v_i in (c) and the chain i in (b) corresponds to the e_i in (c).

The extraction algorithm is shown in algorithm 1. In order to improve the quality of identified lexical chains, a threshold γ is set to filter those invalid words whose generating probability of sampling topics in the document is lower than γ.

$$p(w,d \mid z) \approx p(z \mid d)p(w \mid z) > \gamma \qquad (1)$$

The threshold γ is essential for the quality of lexical chains, which directly impacts on the performance of the model. Detailed analysis for the threshold γ will be given in section 4.5.

3.2 Hypergraph creation

A hypergraph $H = (V, E)$ is a generalization of a graph whose edge can connect more than two vertices. Just as graphs represent many kinds of information in mathematical and computer science problems, hypergraphs also arise in important practical problems, including circuit layout, boolean satisfiability, numerical linear algebra, complex network and article co-citation, etc.

Figure 2 shows an example of hypergraph creation in our model. We represent each context instance as a vertex and connect those context instances with a lexical chain across them by a hyperedge. A hyperedge weight equals to the weight of the corresponding lexical chain, defined as follows:

$$w(e) = \frac{\sum_{w_i \in C} p(z \mid \mathrm{d}_i)p(\mathrm{w}_i \mid z)}{|C|} \qquad (2)$$

where lexical chain C corresponds to hyperedge e, $|C|$ is the number of words in C, and z is the sampling topic of C.

3.3 Hypergraph clustering

For hypergraph clustering, the hypergraph is usually transformed into induced graph. There are two transformation strategies: one is vertex expansions (2006; Zhou et al., 2006), i.e., clique expansion or star expansion, in which a hyperedge is transformed into a clique; the other is called hyperedge expansion (Pu and Faltings, 2012) based on a network flow technique, in which hyperedges are projected back to vertices through the adjacency information between hyperedges and vertices.

Hypergraph clustering algorithm can be divided into two classes: one is based on minimal normalized cut, and the other is based on maximal density. We use three general hypergraph clustering algorithms to identify the context instance clusters. The three algorithms are simply shown in figure 3 and described as the follows.

1) Normalized Hypergraph Cut (NHC)

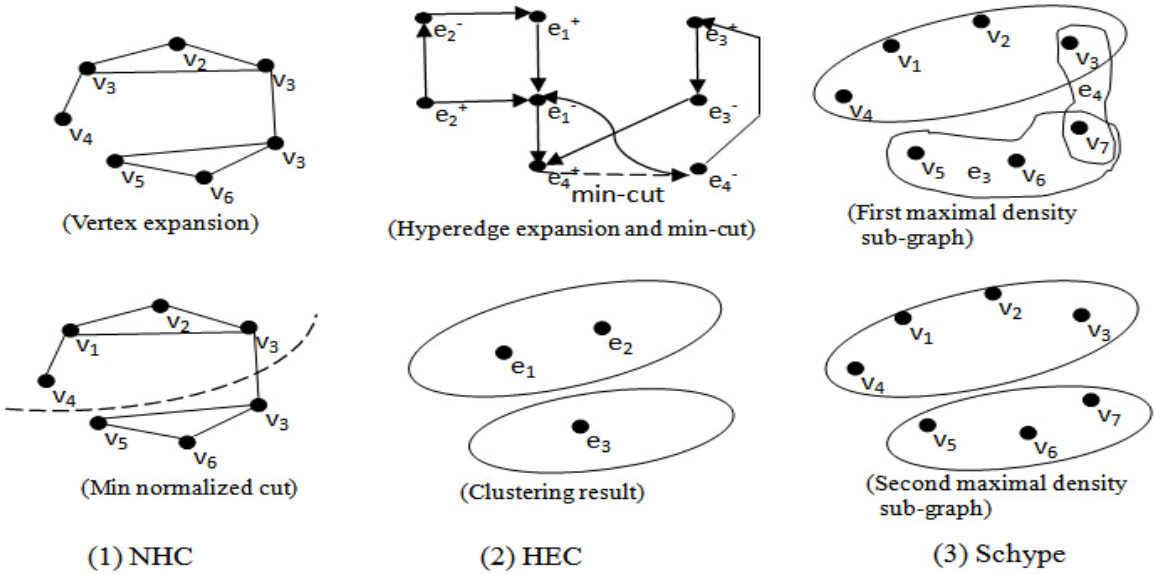


Figure 3. The clustering processes of three algorithms for the hypergraph created in figure 2.

The NHC algorithm (Zhou et al., 2006) is a typical approach based on vertex expansion. The objective is to obtain a partition in which the connection among the vertices in the same cluster is dense while the connection between two clusters is sparse. The main steps include transforming the hypergraph into an induced graph first, and then adopting the normalized Laplacian to spectral partitioning.

2) Hyperedge Expansion Clustering (HEC)

Some works (e.g., Shashua et al., 2006; Bulo and Pelillo, 2012) have shown that the pairwise affinity relations after the projection to the induced graph would introduce information loss, and working directly on the hypergraph could produce better performance.

The hyperedge expansion works as follows. It constructs a directed graph $G = (V, E)$ that includes two vertices $e+$ and $e-$ for each hyperedge e in the original hypergraph. Note that the vertices in G correspond to the hyperedges, but not the vertices in the original hypergraph. A directed edge is placed from $e+$ to $e-$ with weight $w(e)$ where w is the weighting function in the hypergraph. For every pair of overlapping hyperedges e_1 and e_2, two directed edges (e_1^-, e_2^+) and (e_2^-, e_1^+) are added to G with weights $w(e_2)$ and $w(e_1)$. After hypergraph expansion, it adopts spectral method for clustering.

3) Schype

The Schype (Michoel and Nachtergaele, 2012) is a maximal density cluster algorithm. According to the generalization of the Perron-Frobenius theorem, there exists a unique, positive vector, called the dominant eigenvector, over the set of vertices of the hypergraph, which produces a maximal density sub-graph with linear time. The procedure is as follow:

i) Finding maximal density sub-graph by computing the dominant eigenvector.
ii) Removing all vertices and hyperedges of the sub-graph from hypergraph.
iii) Repeating above steps until no vertex in hypergraph occurs.

This algorithm tends to generate many fine-grained clusters. We follow Tan and Kumar (2006) to merge clusters using two measures: cohesion and separation. The cohesion of a cluster C_i is defined as:

$$cohesion(\mathrm{C}_i) = \frac{\sum_{x \in C_i, y \in C_i} \#(e \mid x, y \in e)}{|\mathrm{C}_i|} \qquad (3)$$

where $\#(e \mid x, y \in e)$ is the number of hyperedges containing nodes x and y in C and $|C_i|$ is the number of vertices in C_i. Separation between two clusters C_i, C_j is defined as:

$$separation(\mathrm{C}_i, \mathrm{C}_j) = 1 - \left(\frac{\sum_{x \in C_i, y \in C_j} \#(e \mid x, y \in e)}{|\mathrm{C}_i| \times |\mathrm{C}_j|}\right) \qquad (4)$$

We merge cluster pairs with high cohesion and low separation. The intuition is that context instances in such pairs will maintain a relatively high degree of semantic similarity. High cohesion is defined as greater than average cohesion of all clusters. Low separation is defined as a reciprocal relationship between two clusters: if a cluster C_i has the lowest separations to a cluster C_j and C_j has the lowest separation to C_i, then the two (high cohesion) clusters are merged. This merging process is iterated until it converges.

4 Experiment and Evaluation

4.1 Dataset

Our WSI evaluation is based on the dataset provided by the SemEval-2013 shared 13th task. Test data was drawn from the Open American National Corpus (OANC) (Ide and Suderman, 2004) across a variety of genres and from both the spoken and written portions of the corpus. It consists of 4,806 instances of 50 target words: 20 verbs, 20 nouns and 10 adjectives. Due to the unsupervised nature of the task, participants were not provided with sense-labeled training data. However, WSI systems were provided with the ukWac corpus (Baroni et al., 2009) to use in inducing senses. Additionally, we used the SemEval-2013 lexical trial data sets as development sets to tune parameters.

4.2 Implementation details

The training data is extracted from uKWac corpus. For each target word, we extracted 10K context instances, and each instance is a sentence window containing the target word. Additionally, we randomly selected 10K sentences as common auxiliary corpus, including none of the target word. The training data are tagged with POS tags and lemmatized with TreeTagger (Schmid, 1994). Removing stop words, nouns are taken as features. Meanwhile, we also removed words that co-occur with the target of word less than 50 times over the whole ukWac data.

The training data in the model contains 20K instances: 10K instances of target word, 10K auxiliary instances. Specifically, we used the JGibbLDA[1] framework for topic model estimation and inference, and examined the following LDA parameters: number of topics K, dirichlet hyperparameters for document-topic distribution α and topic-term distribution β. We tested combinations in the ranges K=1000, 1500, 2000, α=5/K..50/K and β=0.001..0.1. The highest performance of the WSI system was found for K = 2000, α =0.025, β = 0.001. Similar to tuning the dirichlet hyperparameters of LDA, the best parameter γ in lexical chain extraction is 0.0001 in the ranges γ =0.01.. 0.000001.

We adopt the three clustering algorithms to cluster hypergraph[2]. The number of clusters is set as 10 for NHC and HEC, while Schype algorithm generates the number of the clusters (but requiring the edge-vertex ratio to be pre-defined), whose average number of senses is 31.8 after clusters are merged. Additionally, For Schype algorithm, we used the default values of parameters, except that the "min-clustscore" parameter, a minimal score to output a cluster, being tuned to 0.1.

The sense inventory acquired from the induction step can be used for disambiguation of individual instances. Each sense is represented as a vector, whose element is a word and the value of element is co-occurrence frequency with target word in the training set. Each test instance is also represented as a vector. The similarity between the instance and the induced sense is computed by using cosine function. For each test instance, it is compared with each sense separately, and finally the sense is selected if the cosine value is greater than a certain threshold λ. In experiment, λ is 0.04 for NHC and HEC, and is 0.1 for schype.

4.3 Evaluation measures

Evaluation in the SemEval-2013 WSI task can be divided into two categories:

1. A traditional WSD task for unsupervised WSD and WSI systems,
2. A clustering comparison setting that evaluates the similarity of the sense inventories for WSI systems.

[1] http://sourceforge.net/projects/jgibblda/

[2] The Hypergraph Analysis Toolbox (HAT) for NHC and HEC: http://lia.epfl.ch/index.php/research/relational-learning and the Schype's code: http://www.roslin.ed.ac.uk/tom-michoel/software/

In the first evaluation, we adopt a WSD task with three objectives: (1) detecting which senses are applicable, (2) ranking senses by their applicability, and (3) measuring agreement in applicability ratings with human annotators. Each objective uses a specific measurement:

i): Jaccard Index: given two sets of sense labels for an instance, X and Y, the Jaccard Index is used to measure the agreement: $\frac{X \cap Y}{X \cup Y}$. The Jaccard Index is maximized when X and Y use identical labels, and is minimized when the sets of sense labels are disjoint.

ii): Positionally-weighted Kendall's τ similarity: for graded sense evaluation, we consider an ranking scoring based on Kumar and Vassilvitskii(2010), which weights the penalty of reordering the lower positions less than the penalty of reordering the first ranks.

iii): Weighted Normalized Discounted Cumulative Gain (WNDCG): NDCG (Moffat and Zobel, 2008) normally compares the rankings of two lists. It is extended to weighting the DCG by considering the relative difference in the two weights.

Because the induced senses will likely vary in number and nature between systems, the WSD evaluation has to incorporate a sense alignment step, in which it performs by splitting the test in-stances into two sets: a mapping set and an evaluation set. The optimal mapping from induced senses to gold-standard senses is learned from the mapping set, and the sense alignment is used to map the predictions of the WSI system to pre-defined senses for the evaluation set. The particular split we use to calculate WSD effectiveness in this paper is 80%/20% (mapping/test), averaged across 5 random splits.

In the clustering evaluation, similarity between participant's clusters and the gold standard clusters is measured by way of two metrics.

i): Fuzzy Normalised Mutual information (NMI): it extends the method of (Lancichinetti et al., 2009) to compute NMI between overlapping clusters. Fuzzy NMI captures the alignment of the two clusters independent of the cluster sizes and therefore serves as an effective measure of the ability of an approach to accurately model rare senses.

ii): Fuzzy B-Cubed: it adapts the overlapping B-Cubed measured defined in Amigo et al. (2009) to the fuzzy clustering setting, and provides an item-based evaluation that is sensitive to the cluster size skew and effectively captures the expected performance of the system on a dataset where the cluster distribution would be equivalent.

4.4 Results

We compared our models with four baselines and three benchmark systems from the SemEval-2013 task. Four baselines are described as follows.

- Baseline MFS — most frequent sense baseline, assigning all test instances to the MFS in the test data (regardless of what applicability rating it was given).
- One sense — labels each instance with the same induced sense.
- One sense per instance (1clinst) — labels each instance with its own induced.
- Baseline Random-n — randomly assigns each test instance to one of n randomly selected induced senses, where n is the number of senses for the target word in WordNet 3.1.

Three benchmark systems as the following are those which achieved better results in the original SemEval-2013 task.

- AI-KU is based on a lexical substitution method.
- UoS uses dependency-parsed features from the corpus, which are then clustered into senses using the MaxMax algorithm (Hope and Keller, 2013).
- Unimelb is a non-parameter topic model which uses a Hierarchical Dirichlet Process (Teh et al., 2006) to automatically infer the number of senses from contextual and positional features.

4.4.1 Supervised evaluation

In the supervised evaluation, the automatically induced clusters are mapped to gold standard senses, using one part of the test set. The obtained mapping is used to label the other part of test set with gold standard tags, which means that the methods are evaluated in a standard WSD task. In experiment, we follow the 80/20 setup: 80% for mapping and 20% for test.

Table 1 shows the results of our systems on test data using all instances (including verbs, nouns and adjectives) for the WSD evaluation. As in previous WSD tasks, the MFS baseline on Jaccard Index measures is quite competitive, outperforming all systems in detecting which senses are applicable.

System	JI-F1	WKT-F1	WNDCG-F1
NHC	0.325	0.692	0.375
HEC	0.327	0.693	**0.376**
Schype	**0.376**	**0.753**	0.345
AI-KU	0.244	0.642	0.332
UNIMELB	0.213	0.62	0.371
UoS	0.232	0.625	0.374
MFS	0.455	0.465	0.339
One sense	0.192	0.609	0.288
1c1inst	0.0	0.095	0.0
Random-n	0.29	0.638	0.286

Table 1. The supervised results over the SemEval-2013 dataset.

System	FNMI	FBC
NHC	0.046	0.406
HEC	0.037	0.400
Schype	0.042	0.377
AI-KU	0.039	0.451
UNIMELB	**0.056**	**0.459**
UoS	0.045	0.448
MFS	-	-
One sense	0	0.632
1c1inst	0.071	0
Random-n	0.016	0.245

Table 2. The unsupervised results over the SemEval-2013 dataset.

However, most systems in this task were able to outperform the MFS baseline in ranking senses and quantifying their applicability.

On the other hand, it also indicates that our three systems achieve better or comparable scores And the Schype gets the highest scores in detecting senses and ranking senses over all systems.

4.4.2 Unsupervised evaluation

Unsupervised evaluation aims to measure the similarity of the induced sense inventories for WSI systems. Unlike supervised metrics, it avoids potential loss of sense information since this setting does not require any sense mapping procedure to convert induced senses to WordNet senses.

Table 2 shows the performance of our systems, benchmarks and baselines. It shows that the NMI-measure is biased towards the **one sense per instance** baseline and the FBC-measure **one sense** baseline. However, systems are capable of performing well in both the Fuzzy NMI and Fuzzy B-Cubed measures, thereby avoiding the extreme performance of either baseline. Generally, the performance of our model gets balanced scores.

4.5 Discussion

Topic models, such as LDA and HDP (Brody and Lapata, 2009; Lau et al., 2012), have been successfully adopted for WSI, in which one topic is viewed as one sense. Our work is motivated by lexical chain that represents the intrinsic semantic relatedness among context instances on the viewpoint of linguistics. Topic model is used to find lexical chains which are interpreted as topics. We have compared the Unimelb (Lau et al., 2013), a HDP topic model, with our model in the experiments. Additionally, we also follow Lau et al. (2012) to train a LDA model with a fixed number of topics based on our training data for WSI[3]. Table 3 shows the supervised result compared to the Schype.

These experiments show promising performance for our model, which captures richer semantic relatedness by using lexical chains. Lexical chains play a key role for the performance of our model. Intuitively, when lexical chains are too long, the higher-order relatedness would be mixed with some noises, while when lexical chains are too short, some higher-order relatedness will be missed. In order to verify the effectiveness of lexical chains, we tune the parameter γ in lexical chain extraction procedure and the results are shown in Figure 4.

Another issue is the impact of POS labels of word for WSI task. The test data in SemeVal-2013 WSI task contain nouns, verbs and adjectives. We also test the performance based on different POS labels. Table 4 gives the supervised evaluation performance of our three systems on adjectives, verbs and nouns respectively. We found that the performance for adjectives in sense detection is the best, verbs followed and nouns worst, whereas it's reversed in sense ranking. The probable interpretation is

[3] The LDA model parameters are set as follows: K=10, α =0.025, β = 0.001.

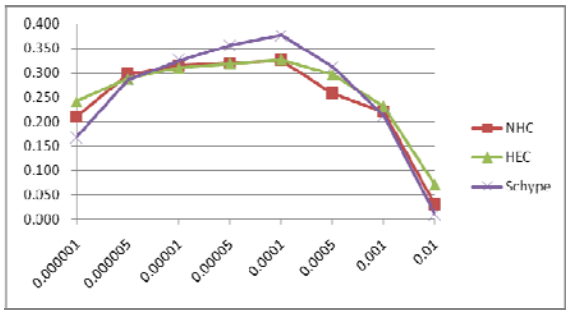


Figure 4. Performance analysis on Jaccard index measure for different threshold parameter γ in lexical chains extraction procedure.

Type	$LDA_{K=10}$	Schype
JI-F1	0.318	0.376
WKT-F1	0.692	0.753
WNDCG-F1	0.334	0.345

Table 3. The supervised results over the SemEval-2013 dataset between $LDA_{k=10}$ and Schype for WSI.

	NHC			HEC			Schype		
POS	**JI**	**WKT**	**WNDCG**	**JI**	**WKT**	**WNDCG**	**JI**	**WKT**	**WNDCG**
nouns	0.306	0.697	0.370	0.313	0.702	0.367	0.363	0.767	0.246
verbs	0.336	0.686	0.374	0.336	0.690	0.380	0.384	0.749	0.245
adjs	0.347	0.697	0.396	0.342	0.678	0.390	0.394	0.733	0.248

Table 4. The supervised performance of three algorithms respectively on nouns, verbs and adjectives.

that adjective's average sense number is the lowest, and the sense granularity is greater than verbs and nouns over the test data[4].

5 Conclusions and future work

In this paper, we present a hypergraph model in which a node represents an instance and a hyperedge represents higher-order semantic relatedness among instances. Compared with other strategies based on binary local comparison, the model captures complex semantic relatedness among the instances from a global perspective.

The evaluation results indicate that our model outperforms or reaches competitive performance comparable to other systems for the SemEval-2013 word sense induction task. Additionally, the experiments also show that both sense number and sense granularity of a target word affect the performance of WSI.

For future work, we would like to explore better ways to extract and evaluate lexical chain for WSI task. In addition, for the three clustering algorithms, they generally require the number of clusters or edge-vertex ratio to be pre-defined, so we will seek more effective hypergraph clustering algorithms to automatically determine the parameters. Finally, the hypergraph model proposed in this work is not specific to the sense induction task, and can be adapted for other applications, such as document classification and clustering, information retrieval, etc.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 61173062, 61373108, 61133012), the major program of the National Social Science Foundation of China (No. 11&ZD189), and the High Performance Computing Center of Computer School, Wuhan University.

Reference

Eneko Agirre and Aitor Soroa. 2007. SemEval-2007 task 02: Evaluating word sense induction and discrimination systems. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 7–12.

Eneko Agirre, David Martnez, Oier López de Lacalle, and Aitor Soroa. 2006. Two graph-based algorithms for state-of-the-art WSD. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language*

[4] In the test data, the average number of senses of nouns, verbs, adjectives respectively is 7.15, 6.85, 5.9 respectively.

Processing, pages 585–593.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Regina Barzilay, Michael Elhadad, et al. 1997. Using lexical chains for text summarization. In *Proceedings of the ACL workshop on Intelligent scalable text summarization*, volume 17, pages 10–17.

Osman Baskaya, Enis Sert, Volkan Cirik, and Deniz Yuret. 2013. AI-kU: Using substitute vectors and co-occurrence modeling for word sense induction and disambiguation. In *Proceedings of SemEval-2013*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of Machine learning research*, 3:993–1022.

Samuel Brody and Mirella Lapata. 2009. Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*.

Stefan Bordag. 2006. Word sense induction: Triplet-based clustering and automatic evaluation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*

Joe Carthy. 2004. Lexical chains versus keywords for topic tracking. In *Computational Linguistics and Intelligent Text Processing*, pages 507–510.

Antonio Di Marco and Roberto Navigli. 2011. Clustering web search results with maximum spanning trees. In *AI* IA 2011: Artificial Intelligence Around Man and Beyond*, pages 201–212.

Beate Dorow and Dominic Widdows. 2003. Discovering corpus-specific word senses. In *Proceedings of the tenth Conference on European chapter of the Association for Computational Linguistics-Volume 2*, pages 79–82.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2009. Investigations on word senses and word usages. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 10–18.

Graeme Hirst and David St-Onge. 1998. Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database*, 305:305–332.

David Hope and Bill Keller. 2013. Maxmax: a graph-based soft clustering algorithm applied to word sense induction. In *Computational Linguistics and Intelligent Text Processing*, pages 368–381.

Nancy Ide and Keith Suderman. 2004. The american national corpus first release. In *LREC*.

Nancy Ide, Tomaz Erjavec, and Dan Tufis. 2001. Automatic sense tagging using parallel corpora. In *NLPRS*, pages 83–90.

David Jurgens and Ioannis Klapaftis. 2013. SemEval-2013 Task 13: Word Sense Induction for Graded and Non-Graded Senses. In *Proceedings of the Seventh International Workshop on Semantic Evaluation* (SemEval 2013).

Ioannis P Klapaftis and Suresh Manandhar. 2007. Uoy: a hypergraph model for word sense induction & disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 414–417.

Andrea Lancichinetti, Santo Fortunato, and János Kertész. 2009. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015.

Jey Han Lau, Paul Cook, and Timothy Baldwin. 2013. unimelb: Topic Modelling-based Word Sense Induction. In *Proceedings of the Seventh International Workshop on Semantic Evaluation* (SemEval 2013).

Jey Han Lau, Paul Cook and Diana McCarthy, David Newman and Timothy Baldwin. 2012. Word sense induction for novel sense detection. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*.

Okumura Manabu and Honda Takeo. 1994. Word sense disambiguation and text segmentation based on lexical cohesion. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*, pages 755–761.

Suresh Manandhar, Ioannis P Klapaftis, Dmitriy Dligach, and Sameer S Pradhan. 2010. SemEval-2010 task 14: Word sense induction & disambiguation. In *Proceedings of the 5th International Workshop on Semantic*

Evaluation, pages 63–68.

Tom Michoel and Bruno Nachtergaele. 2012. Alignment and integration of complex networks by hypergraph-based spectral clustering. *Physical Review E*, 86(5):056111.

Alistair Moffat and Justin Zobel. 2008. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems (TOIS)*, 27(1):2.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational linguistics*, 17(1):21–48.

Roberto Navigli. 2009. Word sense disambiguation: A survey. ACM Computing Surveys (CSUR),41(2):10.

Michael Steinbach Pang-Ning Tan and Vipin Kumar. 2006. *Introduction to Data Mining*. Pearson Addison Wesley.

Patrick Pantel and Dekang Lin. 2002. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD*, pages 613–619.

Li Pu and Boi Faltings. 2012. Hypergraph learning with hyperedge expansion. In *Machine Learning and Knowledge Discovery in Databases*, pages 410–425.

Amruta Purandare and Ted Pedersen. 2004. Word sense discrimination by clustering contexts in vector and similarity spaces. In *Proceedings of the Conference on Computational Natural Language Learning*, pages 41–48. Boston.

Kumar, Ravi and Vassilvitskii, Sergei. 2010. Generalized distances between rankings. In *Proceedings of the 19th international conference on World wide web*, pages 571-580.

Steffen Remus and Chris Biemann. 2013. Three knowledge-free methods for automatic lexical chain extraction. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 989–999, Atlanta, Georgia, June.

Martin Riedl and Chris Biemann. 2012. Sweeping through the topic space: bad luck? roll again! In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*, pages 19–27.

Samuel Rota Bulo and Marcello Pelillo. 2012. A game-theoretic approach to hypergraph clustering.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of international conference on new methods in language processing*, volume 12, pages 44–49.

Hinrich Schütze. 1998. Automatic word sense discrimination. Computational linguistics, 24(1):97–123.

Amnon Shashua, Ron Zass, and Tamir Hazan. 2006. Multi-way clustering using super-symmetric non-negative tensor factorization. In Computer Vision–ECCV 2006, pages 595–608.

Nicola Stokes, Joe Carthy, and Alan F Smeaton. 2004. Select: a lexical cohesion based news story segmentation system. AI Communications, 17(1):3–12.

Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. Journal of the american statistical association, 101(476).

Tim Van de Cruys, Marianna Apidianaki, et al. 2011. Latent semantic word sense induction and disambiguation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*: Human Language Technologies (ACL/HLT), pages 1476–1485.

Jean Véronis. 2004. Hyperlex: lexical cartography for information retrieval. Computer Speech & Language, 18(3):223–252.

Dominic Widdows and Beate Dorow. 2002. A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th International Conference on Computational linguistics-Volume 1*, pages 1–7.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196.

Mehmet Ali Yatbaz, Enis Sert, and Deniz Yuret. 2012. Learning syntactic categories using paradigmatic representations of word context. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 940–951.

Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. 2006. Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in Neural Information Processing Systems*, pages 1601–1608.